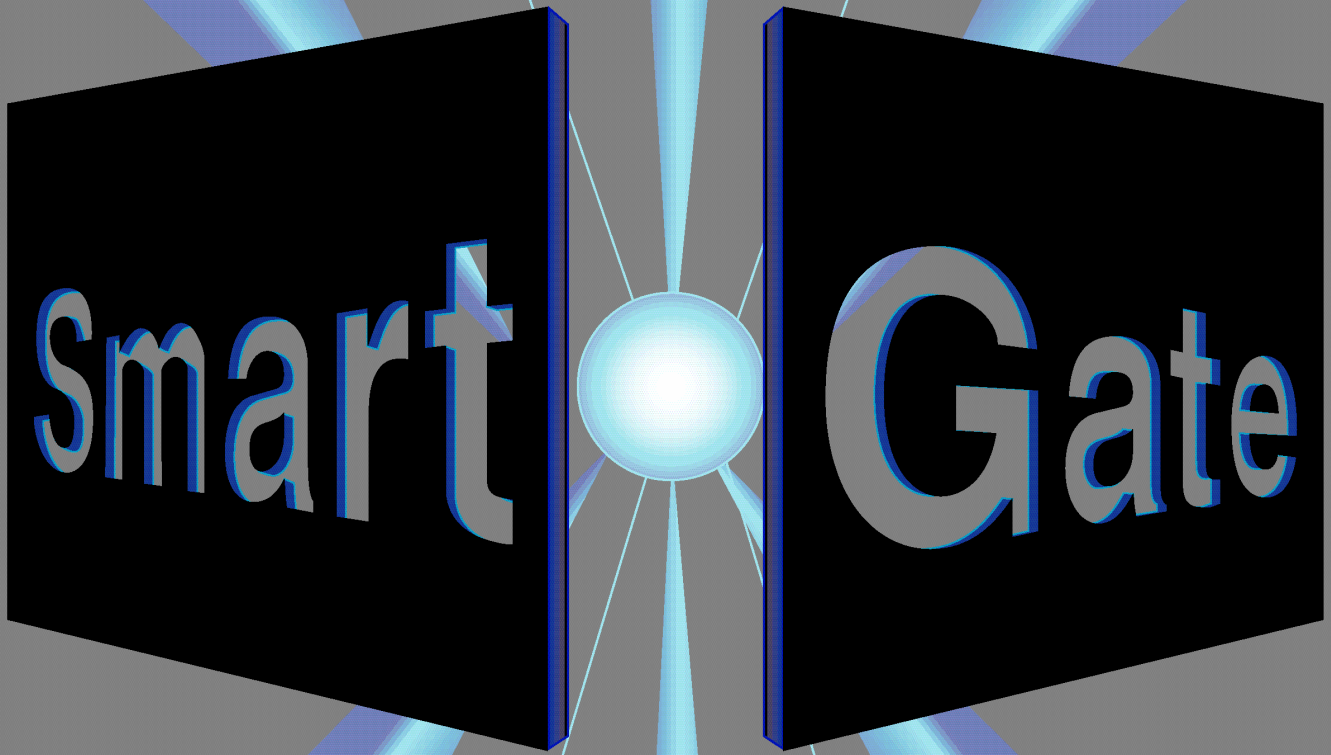


# ***SmartGate***<sup>®</sup>

## Administrator's Guide



***V-ONE***

---

*Security for a Connected World*

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from V-ONE Corporation. No patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this book, V-ONE assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of information contained herein. This book is without warranty of any kind, either expressed or implied. It is further stated that V-ONE is not responsible for any damage or loss to your data or your equipment that results directly or indirectly from your use of this book.

The trademarks mentioned in this book are the property of their respective owners, and may be registered in one or more countries. We strongly advise that you investigate a particular product's name thoroughly before you use the name as your own.

© 2000 V-ONE Corporation  
All Rights Reserved

Published by  
V-ONE Corporation  
20250 Century Boulevard, Suite 300  
Germantown, Maryland 20874  
(301) 515-5200  
(800) 495-VONE (8663)  
(301) 515-5280 (FAX)

V-ONE Corporation Technical Support  
(800) 495-VONE (8663)

(888) 220-VONE (8663)  
(24-Hour Support)

(301) 515-5260  
(International Support)



The public key cryptography software used is proprietary software  
furnished under a license from Baltimore Technologies Limited.

# Contents

<b>Overview .....</b>	<b>13</b>
Purpose of These Guides .....	13
Audience .....	14
Organization of This Guide .....	14
SmartGate Server Installation .....	14
Administration .....	14
SmartGate Server .....	15
Appendices .....	15
Related Materials .....	16
Sources of Help .....	16
Typographic Conventions .....	17
<b>Chapter 1 Introduction to SmartGate .....</b>	<b>19</b>
SmartGate System Components .....	20
SmartGate Server .....	21
SmartPass .....	21
Authentication Methods .....	21
Access Code .....	22
Hardware and Software Requirements for the SmartGate Server .....	22
UNIX Systems .....	22
Intel Systems .....	23
Linux Systems .....	23
Sun SPARC Systems .....	23
Microsoft Windows NT System .....	24
Hardware and Software Requirements for SmartPass .....	24
PC Users .....	24
UNIX Users .....	24
Macintosh Users .....	25
Windows CE Devices .....	25
Pocket PC Devices .....	25
SmartGate Server Version 4.1 New Features .....	25
KRAKit™ (Key Recovery Agent's Kit) .....	26
Acquiring a License .....	27
<b>Chapter 2 Preinstallation of the SmartGate Server Software .....</b>	<b>29</b>
Perform Preinstallation .....	29
Setting Up Your TCP/IP Protocol Properties .....	30
Obtain Your Installation Values .....	31

<b>Chapter 3 Installing the SmartGate Server Software—UNIX .....</b>	<b>35</b>
Installing the Software on Your System .....	35
Upgrading from a Prior Version .....	35
General Installation Instructions .....	36
Running the Installation Script .....	36
Obtaining Your License and Certificate .....	39
Minimal UNIX Configuration for Remote Administration .....	41
Setting Up the SmartGate Server at the UNIX Console .....	42
Regenerating the SmartGate Public/Private Key Pair .....	42
Adding a License or Viewing Your License Key .....	44
Configuring the SmartGate Server Software (sgconf.ini) .....	44
Setting Up the On-Line Registration File (reginfo.dat) .....	45
Branding the On-Line Registration Web Page (sgconf.ini) .....	48
Configuring Access Permission Files (sgate.acl and sweb.acl) .....	49
Adding Administrative Privileges (adm-gw.acl) .....	51
Configuring Single Port Proxy Services .....	53
Configuring SmartGate Extensible Components .....	54
RSA SecurID Authentication .....	54
RADIUS Authentication .....	55
Entrust/Netrust Authentication .....	57
PKI Authentication .....	58
Rebooting Your Computer .....	59
Backing Up SmartGate Configuration Files .....	59
Uninstalling the Software .....	60
<b>Chapter 4 Installing the SmartGate Server Software—Windows NT .....</b>	<b>61</b>
Installing the Software on Your System .....	61
Obtaining Your License and Certificate .....	69
Launching SmartAdmin .....	71
Adding and Removing Services .....	71
<b>Chapter 5 Using SmartAdmin .....</b>	<b>73</b>
Launching SmartAdmin .....	74
Preliminaries for Remote Administration .....	75
Setting Yourself Up as an Administrator .....	75
Basic Usage .....	76
Managing Users .....	77
Add User .....	78
Edit User .....	79
Delete User .....	80
Multiple User Management .....	80
Find and Find Next .....	81
Filter Users .....	81
Common User Management Tasks .....	82



TCP and Web Access Permissions .....	82
Defining Access Permissions .....	83
TCP Access Permissions .....	84
Add/Edit TCP Access Permissions .....	87
Delete TCP Access Permissions .....	88
Filtering .....	89
Web Access Permissions .....	89
Add/Edit Web Access Permissions .....	91
Delete Web Access Permissions .....	91
Filtering .....	92
Setting Up On-Line Registration .....	93
OLR Branding Options .....	94
Assigning Administrative Rights .....	96
Single Port Client .....	97
Configuring Single Port Proxy .....	98
Add/Edit Port Map Rules .....	99
Changing the Default Single Port Proxy .....	100
SmartGate Server .....	100
SmartPass .....	101
Setting Configuration Options .....	102
Access Control Settings .....	102
Use TCP access control (sgateacl) .....	103
Use Web access control (swebacl) .....	103
Server proxy timeout (max_quiet_time) .....	103
Web Denial Server host and port (denial_server) .....	104
Forbidden Web servers (sweb_not_allowed) .....	104
On-Line Registration Settings .....	105
OLR methods (OLRMethod) .....	105
User ID Servers host and port (uid_server) .....	105
User ID Rules File (UidFile) .....	107
New OLR users enabled (online_reg_enable) .....	107
Netrust anonymous registration (anon_reg_allowed) .....	107
OLR data destination host and port (online_reg_service) .....	108
Encryption keyname (keyname) .....	108
Logging .....	108
Reverse DNS lookups (dns_reverse) .....	109
Accounting service host and port (accounting_service) .....	109
Usage service host and port (stat_server) .....	109
Event log service host and port (event_log) .....	110
Debug reporting (debug) .....	110

System Definition Settings .....	111
SmartGate Server name (domainname) .....	111
Port list (PortList) .....	111
UDP port list (UDPPortList) .....	111
Authenticator name (authenticator) .....	111
Inside IP address (InsideIP) .....	111
SG encryption methods (SGEncryptMethod) .....	112
Proxy encryption methods (ProxyEncryptMethod) .....	112
Authentication Settings .....	113
Authentication methods (AuthMethod) .....	113
Remote Authentication Server (sgasrv) .....	113
Authentication Server host .....	113
Authentication client hosts (sgasrv_clients) .....	114
Authentication encryption methods (AuthEncryptMethod) .....	114
Backup server host and port (backup_userdb) .....	114
Access failure retry delay (RETRY_DELAY) .....	115
Trust CA list (TrustedCAList) .....	115
Dynamic Configuration Settings .....	115
Remote Configuration Server (sgccsrv) .....	117
Configuration Server host and port .....	117
Configuration client hosts (sgccsrv_clients) .....	118
Destination Configuration Settings .....	118
SmartGate aware services (SmartGate_aware) .....	118
User info to Web server (UserInfoToWebServer) .....	119
Web servers requiring encrypted tickets (ticket_to_web_server) .....	120
RADIUS Settings .....	121
RADIUS Backend Servers: Host (radius_authsrv[1...5]) .....	121
RADIUS Backend Servers: Secret (radius_authsrv[1...5]_secret) .....	121
RADIUS Backend Servers: Use CHAP (radius_authsrv[1...5]_usechap) .....	121
RADIUS Backend Servers: Wait (radius_authsrv[1...5]_waitfor) .....	122
Time to live (radius_ttl) .....	122
Challenge timeout (radius_challenge_timeout) .....	122
Other Settings .....	122
accesscodedaysvalid (AccessCodeDaysValid) .....	122
krakit_delta_days (KraKit_Delta_Days) .....	123
sdi_timeout (SDI_TIMEOUT) .....	123
sdi_ttl (SDI_TTL) .....	123
sgftp_port_max (sgftp_port_max) .....	123
sgftp_port_min (sgftp_port_min) .....	123
shim_permitexe (shim_permitexe) .....	123
smartwebport (SmartWebPort) .....	123
PKI Administration .....	124

<b>Chapter 6 User Authentication .....</b>	<b>125</b>
SmartGate Authentication Server .....	125
Remote Authentication Server .....	128
Server Redirection Access Permissions .....	128
Backup Server Host .....	129
Using RSA SecurID for User Authentication .....	131
Making SmartGate an ACE Client .....	131
Running the sgsdi Service .....	131
Configuring the SmartGate Server .....	133
sdi_timeout .....	133
sdi_ttl .....	133
Using RADIUS for User Authentication .....	133
Running the sgradius Service .....	133
Configuring the SmartGate Server .....	135
RADIUS Backend Servers: Host (radius_authsrv[1...5]) .....	135
RADIUS Backend Servers: Secret (radius_authsrv[1...5]_secret) .....	135
RADIUS Backend Servers: Use CHAP (radius_authsrv[1...5]_usechap) .....	136
RADIUS Backend Servers: Wait (radius_authsrv[1...5]_waitfor) .....	136
Time to live (radius_ttl) .....	136
Challenge timeout (radius_challenge_timeout) .....	136
Configuring the RADIUS Port .....	136
Configuring the RADIUS Backend Server .....	137
SmartPass/RADIUS Backend Server Interaction .....	137
Using Entrust for User Authentication .....	138
Installing and Configuring the Entrust Software .....	139
Configuring the entrust.ini File .....	139
Installing the SmartGate Server Software .....	140
Microsoft Windows NT SmartGate Server .....	140
UNIX-Based SmartGate Server .....	142
Configuring the SmartGate Server .....	143
Preparing the SmartPass Installation Package for Entrust Authentication .....	144
Configuring setup.ini .....	144
Copying the entrust.ini File .....	145
Preparing the entrust.z Archive .....	145
Viewing a Distinguished Name .....	146
Using PKI Authentication for User Authentication .....	147
Digital Certificates .....	147
Personal Certificates .....	148
Adding CA Certificates to the Trusted CA List .....	148
Command Line Configuration Variables for certmanager.exe .....	148

<b>Chapter 7 On-Line Registration Services .....</b>	<b>149</b>
SmartGate Server Configuration for On-Line Registration .....	149
UID Server for On-Line Registration .....	150
How This Feature Works .....	150
Setting Up Your SmartGate UID Server .....	151
Configuring Your SmartGate Server .....	151
Creating a Rules File .....	153
Registering Your UID Server .....	154
Creating Your Own UID Server Process .....	155
Manual Setup of an HTML Page for On-Line Registration .....	156
Web Server Configuration .....	157
Specification of Mandatory OLR Parameters .....	157
User Inputs .....	158
Specification of Optional OLR Parameters .....	159
Customer Branding .....	159
Creation of a Desktop Shortcut .....	160
Sample Form .....	160
SmartPass Deployability .....	162
Performing OLR Through a Firewall .....	163
<b>Chapter 8 Request for Passive Open (PASV) .....</b>	<b>165</b>
PASV File Transfer Protocol .....	165
Using the FTP Passive Mode With SmartGate .....	165
Setting Up an FTP Client That Supports PASV .....	165
<b>Chapter 9 Oracle .....</b>	<b>167</b>
SmartGate Oracle SQLNet II Proxy .....	167
Setting Up the SmartGate Server for the Oracle SQLNet Proxy .....	167
Removing or Reinstalling the Oracle Service .....	168
Oracle Setup .....	168
Setting Up Access Permissions .....	169
Testing Your Connection .....	169
Single Port Configuration .....	170
Troubleshooting Oracle .....	170
<b>Chapter 10 Using the vplug Proxy .....</b>	<b>171</b>
How vplug Works .....	171
Configuring the netaccess.cf File .....	172
Example of a netaccess.cf File .....	172
Additional Routes .....	173
netaccess.cf Parameters Supported by vplug .....	173
net_list Guidelines .....	175
Using a net_list File .....	175
vplug Options for UNIX .....	176
vplug for Windows NT .....	177

<b>Chapter 11 IPSec .....</b>	<b>179</b>
What is IPSec? .....	179
IPSec's Core Components .....	179
Transport and Tunnel Mode .....	180
ESP - Encapsulating Security Payload .....	180
AH - Authentication Header .....	180
IPCOMP - IP Payload Compression .....	181
NAT - Network Address Translation .....	181
Putting Together ESP, AH, IPCOMP, and NAT .....	182
IPSec Functionality .....	184
Choosing Your IPSec Network Topology .....	185
Routing .....	187
Security Levels for Adapters (Interfaces) .....	189
Setting Up Your TCP/IP Protocol Properties .....	189
Configuring IPSec Settings .....	191
IPSec Server External Interface (ipsec_server_extrn) .....	191
NAT Enabled (NAT) .....	191
NAT Network (NATNet) .....	192
Adapter Security Levels .....	192
Configuring IPSec Channel Types .....	193
Add/Edit IPSec Channels .....	195
Delete IPSec Channels .....	196
Configuring IPSec Access Permissions .....	197
Add/Edit IPSec Path or Include Access Permissions .....	198
Delete IPSec Path Permission .....	200
IPSec DNS Proxy Overview .....	200
Add/Edit IPSec DNS Proxy Access Permissions .....	202
Delete IPSec Access Permissions .....	203
Using the VIPUTIL Utility .....	204
Protocol Number Definitions .....	205
Site-To-Site IPSec .....	208
Operational Description .....	208
Firewall Considerations .....	208
Routing Considerations .....	209
Configuration Information .....	209
The Site-to-SiteTab .....	209
<b>Glossary .....</b>	<b>217</b>
<b>Appendix A SmartGate Server Files .....</b>	<b>227</b>
SmartGate Server Files Detailed Descriptions .....	227
Access Control Lists .....	227
ipsec.acl .....	227
sites.acl File .....	231
Individual Fields .....	231
[next.]* .....	234

sgate.acl .....	237
sgate.dny .....	238
sweb.acl .....	239
sweb.dny .....	240
Database Files .....	241
sgusrdb .....	241
dbrw .....	241
Configuration Files .....	242
adm-gw.acl .....	242
aliases. ....	244
chantype.ini .....	245
sgconf.ini .....	248
reginfo.dat .....	250
netaccess.cf .....	254
net_list .....	254
Optional Server Files for Enhanced Features .....	255
Rules File .....	255
SmartGate Server File Option Descriptions .....	258
sgconf.ini Options .....	258
AccessCodeDaysValid .....	260
accounting_service .....	260
anon_reg_allowed .....	261
authenticator .....	261
AuthEncryptMethod .....	262
AuthMethod .....	262
backup_userdb .....	262
debug .....	263
denial_server .....	263
dns_reverse .....	264
domainname .....	264
event_log .....	265
InsideIP .....	265
ipsec_server_extrn .....	265
Krakit_Delta_Days .....	266
max_quiet_time .....	266
NAT .....	266
NATNet .....	267
OLRAIIOutsideFirewall .....	267
OLRCity .....	267
OLRCompanyName .....	268
OLRCountry .....	268
OLREmail .....	268
OLRPhoneNumber .....	268
OLRStartArgs .....	268
OLRStartDesc .....	269
OLRState .....	269
OLRStreetAddress .....	269

OLRWebPage .....	269
OLRZipCode .....	269
OLRMethod .....	270
online_reg_enable .....	270
online_reg_service .....	271
Port List .....	272
ProxyEncryptMethod .....	272
radius_authsrv[1...5] .....	272
radius_authsrv[1...5]_secret .....	273
radius_authsrv[1...5]_usechap .....	274
radius_authsrv[1...5]_waitfor .....	274
radius_challenge_timeout .....	275
radius_ttl .....	275
RETRY_DELAY .....	275
SDI_TIMEOUT .....	276
SDI_TTL .....	276
sgasrv .....	276
sgasrv_clients .....	277
sgateacl .....	278
sgccsrv .....	278
sgccsrv_clients .....	279
SGEncryptMethod .....	279
sgevent_logging .....	280
sgftp_port_max .....	280
sgftp_port_min .....	281
shim_permitexe .....	281
SmartGate_aware .....	281
SmartWebPort .....	282
stat_server .....	282
swebacl .....	282
ticket_to_web_server .....	283
TrustedCAList .....	283
UDPPortList .....	284
UidFile .....	284
uid_server .....	284
UserInfoToWebServer .....	285
<b>Appendix B Services .....</b>	<b>287</b>
Accounting Service .....	287
Denial Server .....	289
Switching Between the Production and Debug Logs .....	290
How To FTP .....	291
<b>Appendix C ACL Wildcarding .....</b>	<b>293</b>
DNS Wildcarding .....	294
Valid and Invalid DNS Wildcarded Addresses .....	294



IP Wildcarding .....	295
Subnet-Mask IP Wildcarding .....	295
Significant-Bit IP Wildcarding .....	296
*-Character IP Wildcarding .....	296
Valid and Invalid *-Character IP Wildcarded Addresses .....	296
Port Wildcarding .....	297
Order of Evaluation .....	297
ACL Specification .....	298
Grammar for the sgate.acl File .....	298
Grammar for the sweb.acl File .....	298
Grammar for One Access Permission Rule in sgate.acl .....	298
Grammar for One Access Permission Rule in sweb.acl .....	298
Grammar Details for Handling Wildcards in ACLs .....	299
Pattern Matching Actual Destinations with Wildcarded Destination ACLs .....	300
Validating ACLs Which May Contain Wildcards .....	300
Dealing with Wildcarded Ports .....	300
<b>Index .....</b>	<b>301</b>

# Overview

**NOTE:** IPSec is only available with SmartGate 4.0 and later running on a Microsoft Windows NT server.

V-ONE Corporation's SmartGate is a leading [client/server virtual private network \(VPN\)](#) security software system with a clearly defined mission: to provide enterprise-level [security](#) for network-based users to private information and private [TCP/IP](#) application services. SmartGate, distributed worldwide, provides strong user [authentication](#), authorization, management, accounting, encryption, key distribution, and [proxy](#) capabilities. SmartGate also provides driver-level [IPSec](#) transport functionality in addition to our traditional proxy capabilities. IPSec is a method of encapsulating IP packets (not just TCP sessions) to encrypt and protect data while enroute, not just from modification, but also from examination.

Using SmartGate, businesses, public organizations, and governmental agencies of all sizes can deliver fast, cost-effective security solutions to communities of Internet, intranet, and extranet users.

## Purpose of These Guides

These guides are designed for the [SmartGate Server administrator](#) to manage both locally and remotely the [SmartGate Server](#) and their [end users'](#) access.

## Audience

This guide is intended for use by the SmartGate administrator. It assumes that you have experience as a network administrator working with operating systems running UNIX or Microsoft® Windows® NT, and that you are familiar with [firewalls](#) and the Internet.

## Organization of This Guide

Chapter 1      **Introduction to SmartGate**  
Basics of SmartGate System and key SmartGate features.

## SmartGate Server Installation

Chapter 2      **Preinstallation of the SmartGate Server Software**  
General instructions for the preinstallation of the SmartGate Server software and definitions of key terms.

Chapter 3      **Installing the SmartGate Server Software - UNIX**  
Instructions for installing and locally configuring the SmartGate Server software running on a UNIX-based Server.

Chapter 4      **Installing the SmartGate Server Software - Windows NT**  
Instructions for installing and minimal configuration of the SmartGate Server software running on a Windows NT Server.

## Administration

Chapter 5      **Using SmartAdmin**  
Instructions on using [SmartAdmin](#) to configure and manage the SmartGate Server, including user information, [access permissions](#), and OLR setup.

## SmartGate Server

- Chapter 6     **User Authentication**  
A detailed description of the Authentication Server (`sgasrv`) and its features, including user authentication, access control, its cooperative role during OLR, the remote authentication server, server redirection, and backup server, as well as detailed descriptions of RSA SecurID, RADIUS, and Entrust authentication. Instructions for installing PKI Server certificates are also described.
- Chapter 7     **On-Line Registration Services**  
Instructions on optional SmartGate services including the UID Server, manual setup of an HTML page for OLR, and use of a proxy to navigate through a firewall.
- Chapter 8     **Request for Passive Open (PASV)**  
Instructions on configuring the PASV option.
- Chapter 9     **Oracle**  
Instructions for configuring Oracle.
- Chapter 10    **Using the `vpplug` Proxy**  
Instructions for the `vpplug` Proxy which provides virtual hosting for generic TCP connections.
- Chapter 11    **IPSec**  
A detailed description of the driver-level IPSec transport functionality available on a Microsoft Windows NT SmartGate Server version 4.0 and instructions on how to install and use IPSec.

## Appendices

- Appendix A   **SmartGate Server Files**  
Detailed descriptions and examples of the major SmartGate Server files and their configurable options.
- Appendix B   **Services**  
Descriptions of some of the services available with SmartGate.
- Appendix C   **ACL Wildcarding**  
Description of the Access Control Lists (ACL) wildcarding process available with SmartGate.

## Related Materials

Depending on your system configuration and the features you plan to implement within your organization, you should also refer to the following information sources:

- *The SmartPass Administrator's Guide*
- *The SmartGate With Netrust Authentication Guide*
- *The Air SmartGate Administrator's Guide*
- *The KRAKit™ (Key Recovery Agent's Kit) Guide*
- The vplug Proxy. For related information, list the man page for **ifconfig**
- The Oracle SQLNet Proxy documentation
- Your RADIUS Backend Server documentation
- RSA Security, Inc., ACE/Server documentation
- Entrust documentation
- Netrust documentation

## Sources of Help

If you experience any problems with your installation or if you need assistance, please contact V-ONE Technical Support at (888) 220-8663 or (301) 515-5260 for international calls. You may also contact V-ONE Customer Care Support through e-mail at [customercare@v-one.com](mailto:customercare@v-one.com).

## Typographic Conventions

This guide uses the following typographic conventions to distinguish user- and system-generated syntax, to identify software components, and to display precautionary messages and other guidance for the administrator/user.

To Show...	We use...	Example
Required data or command keywords	Minion Web bold	type: <b>vplug</b>
Data or prompts displayed by the system	Courier	Password
Filenames, directory names, program names, hostnames, or IP addresses in text	Courier	the sgconf.ini file
Variable parameters or options	Italic	<b>sagadm -enable <i>userid</i></b>
Choice of options	(pipe; not typed by user)	sgateacl=yes no
Keyboard keys	Minion Web small caps	Press ENTER
Items of importance that facilitate installation and use of the software	<b>NOTE:</b>	<b>NOTE:</b> This process is controlled by values in the sgconf.ini file ...
Issues or instructions that, if not acknowledged or adhered to, could cause a loss of data or pose a serious threat to the security of your system	<b>WARNING!</b>	<b>WARNING!</b> You must enter an Access Code within 30 seconds ...
Limitation on use of the software by the Macintosh Operating System	<b>Macintosh USERS:</b>	<b>Macintosh USERS:</b> Oracle is not supported by the Macintosh OS





# Chapter 1

# Introduction to SmartGate

**NOTE:** IPSec is only available with SmartGate 4.0 and later running on a Microsoft Windows NT server.

**NOTE:** For detailed information on IPSec, see [Chapter 11, "IPSec."](#)

The SmartGate System provides application-level data security for public and private networks through the integration of:

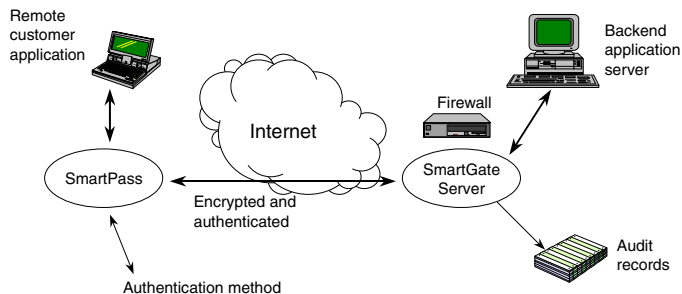
- Two-factor user authentication, using both physical smart cards and virtual smart cards (soft tokens).
  - An authentication token that the user *has*
  - An [Access Code](#) that the user *knows*
- Access control based on authenticated user identification, independent of IP address.
- Electronic distribution, with user On-Line Registration (OLR).
- Driver-level IPSec transport functionality available on a Microsoft Windows NT server. IPSec is a method of encapsulating IP packets to encrypt and protect data from modification enroute. Encryption and data protection are proxy based for Solaris, Linux, Macintosh, Windows CE, and Pocket PC platforms.
- Application- and system-independent TCP/IP interoperability. SmartGate is compatible with previous systems, mainframes, token rings, ethernet, independent LANs/WANs, and most TCP/IP applications.
- Strong, transparent security for most TCP-based applications.
- Server software that supports Microsoft Windows NT and UNIX operating systems and client software that supports Windows, UNIX, Macintosh, and Windows CE operating systems.
- Dynamic Configuration that virtually eliminates the need for local configuration by the end user. When launched, SmartPass immediately contacts every SmartGate Server for

which the user has an [authentication key](#), requests a current list of the user's access permissions, and establishes secure connections between the user's workstation and the available SmartGate communities.

- Multiple Web and TCP access permission assignment using access control lists (ACL) wildcarding.
- An optional strong encryption standard. [Triple DES](#) (Data Encryption Standard) encryption is available for an even higher level of security.

## SmartGate System Components

SmartGate consists of the following basic components (Figure 1-1):



- A SmartGate Server.
- [SmartPass](#) software that resides on a user's personal computer.
- An authentication method.
- An Access Code for each authentication token.

**NOTE:** For detailed information on ACL wildcarding see [Appendix C, "ACL Wildcarding."](#)

**Figure 1-1**  
**SmartGate System**

**NOTE:** For detailed information on SmartPass, see the *SmartPass Administrator's Guide* for more information.

**Macintosh USERS:** Physical smart cards are not supported by SmartPass for Macintosh.

**NOTE:** The title “MCOS” includes support for both MCOS and MCOS-B smart cards. Use the MCOS card that is pertinent to your network configuration.

## SmartGate Server

The SmartGate Server authenticates users and manages access control. It contains an access control database that maps the relationships between users and designated application services. The SmartGate Server allows administrators to assign each user to a [SmartGate group](#), and to apply user- and group-level access privileges to the services that each user may access. On a Microsoft Windows NT Server, they can also assign user- and group-level IPsec access permissions with specific channel types, utilizing the IPsec transport functionality. For example, a site may have users in a “customer” group whose members are permitted access only to an SQL service behind the firewall. Other users may be in the “staff” group, which can access Telnet, POP3, [SMTP](#), and [FTP](#) services. Yet, certain users working on a specific project, the “projectX” group, may have access only to an FTP site.

## SmartPass

SmartPass is SmartGate’s client software. It runs on the end user’s computer. It manages user authentication and data stream encryption between the user’s computer and the SmartGate Server.

## Authentication Methods

SmartGate’s authentication system supports soft tokens and ISO-standard (physical) smart cards for authentication. A physical smart card is used in conjunction with a [smart card reader](#) connected to the user’s computer. These methods are:

1. [FIPS token](#) (FIPS 140-1 compliant), a virtual token on a smart card or hard drive
2. [VCAT token](#), a virtual token on a smart card or hard drive
3. [PCAT reader](#) (accepts [MCOS](#) or [STARCOS](#) physical smart cards)
4. [Smarty reader](#) (accepts [MCOS](#) or [STARCOS](#) physical smart cards)
5. [CHIPDRIVE external reader](#) (accepts [STARCOS](#) physical smart cards)

Soft token information may be stored on either the computer’s hard drive or a removable disk. The user’s SmartGate authentication key is stored on either the physical smart card or soft token and in the SmartGate Server’s user database.

SmartGate also supports third-party authentication methods:

- [RSA SecurID authentication](#)
- [RADIUS authentication](#)
- [Netrust authentication](#)
- [Entrust authentication](#)
- [PKI authentication](#)

## Access Code

Each time the user accesses a secure service, an Access Code, similar to a PIN code on an ATM card, is required to unlock the authentication key stored on the user's smart card.

Features of the Access Code include:

- The user can request that SmartPass “remember” the Access Code for up to 999 minutes. This is the time allowed for session inactivity before SmartPass prompts you for your Access Code.
- The user can suspend his/her Access Code through the SmartPass User Interface's “Forget” option. The “forgotten” Access Code must be reentered when the secured session is resumed.

## Hardware and Software Requirements for the SmartGate Server

The SmartGate Server software, version 4.x, is available for both UNIX and Microsoft Windows NT operating systems.

### UNIX Systems

SmartGate for UNIX runs on three hardware/operating system platforms:

- Intel Systems running [BSD](#)/OS 3.1, 4.0 or 4.1
- Linux Systems running RedHat [Linux](#) 6.0, 6.1, or 6.2
- Sun SPARC Systems running Solaris 2.6, 7, or 8

System requirements for each operating system are given below.

## Intel Systems

Hardware and software requirements for Intel systems include:

- BSD/OS 3.1, 4.0, or 4.1
- 20 megabytes (MB) minimum of free hard disk space
- 128 MB of random access memory (RAM) or higher
- Pentium II, Celeron, or equivalent processor at 350 megahertz (MHz)
- The following BSD/OS software and packages are required for a minimum installation:
  - All required packages
  - Additional `/usr`
  - Networking
  - Man pages (recommended; not required)
- 2 [network adapter cards](#)

## Linux Systems

Hardware and software requirements for Linux operating systems include:

- Pentium II, Celeron, or equivalent processor at 350 MHz running RedHat Linux 6.0, 6.1, or 6.2
- 20 MB minimum of free hard disk space
- 128 MB of RAM or higher
- All required software packages
- 2 identical network adapter cards

## Sun SPARC Systems

Hardware and software requirements for SPARC systems include:

- Sun Solaris 2.6, 7, or 8
- 20 MB minimum of free hard disk space
- 128 MB of RAM or higher
- All required software packages
- 2 network adapter cards

## Microsoft Windows NT System

Hardware and software requirements for Microsoft Windows NT systems include:

- Microsoft Windows NT operating system, version 4.0, service pack 5 or 6a, running the NT file system (NTFS)
- Pentium II, Celeron, or equivalent processor at 350 MHz
- 128 MB of RAM or higher
- 10 MB of free hard disk space
- 1 network adapter card (2 cards recommended)

## Hardware and Software Requirements for SmartPass

To install SmartPass, most hardware and software requirements depend upon the operating system. However, all user computers must have:

- Internet access
- Connection to a network using TCP/IP protocol
- The appropriate SmartPass software

### PC Users

- Microsoft Windows 95, osr2, 95b, 98, 98SE, or Windows NT Workstation, version 4.0, with service pack 5 or 6a
- Microsoft Windows 2000 (proxy through localhost only NO shim support)
- 4 MB of free hard disk space
- Netscape Navigator 4.5, 4.51, 4.61, 4.7, 4.72, 4.73, or Microsoft Internet Explorer 5.00, 5.01 (5.01 with service pack 1)

### UNIX Users

- A computer with either Sun SPARC Systems running Solaris 2.6 or later; or an Intel (or compatible) system running RedHat Linux 6.0, 6.1, or 6.2
- 5 MB minimum of free hard disk space
- A suitable UNIX Web browser (must support forms)

**NOTE:** You can check which service pack is installed on your Windows NT by clicking **Start**, **Run**, and typing **winver**.

**NOTE:** Higher disk usage may be required on high traffic servers due to logging purposes.

**NOTE:** For an updated list of IPSec-compatible network adaptor cards, see: <http://www.v-one.com/techsupport/tn001.htm/>.

**NOTE:** PKI authentication requires License code 54.

## Macintosh Users

- An Apple or other Macintosh OS-compatible Power PC computer
- 1 MB of free hard disk space
- Macintosh OS Version 8.1 or later
- Open Transport 1.3 or later
- Open Transport TCP/IP (MacTCP is NOT supported)

## Windows CE Devices

The SmartPass CE software supports the following devices:

- Handheld PC (SH3 and MIPS)
- Handheld PC Professional Edition (SH3, SH4, MIPS, ARM, and StrongARM)
- Palm-size PC (SH3 and MIPS)

## Pocket PC Devices

The SmartPass Pocket PC software supports the following devices:

- Hewlett-Packard Jornada 545
- Casio Cassiopeia E-115

## SmartGate Server Version 4.1 New Features

- SmartGate includes PKI authentication. This method of authentication allows SmartPass users to use a PKI certificate in place of other V-ONE tokens to be authenticated by a SmartGate Server. This feature makes it convenient for those who already have a PKI-based certificate to use that certificate for user authentication in the SmartGate System.
- Site-to-Site IPSec on Microsoft Windows NT. This feature enables the SmartGate administrator to set up VPN connections between entire networks via an IPSec tunnel from a SmartGate Server installed on one network to a SmartGate Server installed on another network.



- Deployability enhancements have been made for SmartPass users. The SmartGate Server will listen to incoming connections on all ports listed in the SmartGate configuration file (`sgconf.ini`) and SmartPass will perform OLR on the corresponding ports listed in the `setup.ini` file. Once a port has been found acceptable, that port will be stored into the registry. All future connections will traverse only on the stored port.
- Smart card removal detection. SmartPass detects removal or replacement of a smart card from a smart card reader. When SmartPass detects removal or replacement of the smart card, it stops all current connections, flushes all cached information, and shuts down the SmartPass session.

## KRAKit™ (Key Recovery Agent's Kit)

Approved by the Bureau of Export Administration (BXA), V-ONE's key recovery feature KRAKit™ permits customers to use strong encryption (up to 168 bits) for any application in virtually any country while enabling an organization to keep control of their own session encryption keys. This approval now enables organizations to deploy 168-bit encryption with V-ONE's SmartGate VPN for secure communications. Strong encryption can be used domestically as well as internationally (anywhere other than the seven countries embargoed by the U.S. Government—Iran, Iraq, Syria, Sudan, Cuba, North Korea, Libya) without the need for cumbersome third party or escrow agent key recovery processes as long as the customer agrees to manage his own keys.

Trusted First Party (TFP) is the name of V-ONE's key recovery methodology that allows for the recreation of session encryption keys while ensuring that the customer always controls these keys. TFP is comprised of SmartGate, V-ONE's VPN product, along with the session key recreation capabilities of SmartGate's KRAKit feature.

**NOTE:** For complete information on KRAKit, please refer to V-ONE's *KRAKit Guide*.

**WARNING!** SmartGate will not run without a valid License Key and certificate.

## Acquiring a License

To acquire a [License Key](#) and certificate for your SmartGate Server, you must either use V-ONE's licensing Web site at <http://license.v-one.com/license/html/gtsmartgate.html> or call V-ONE Corporation at (800) 495-VONE(8663) or (301) 515-5260 from an international location. You will need a [Customer ID](#) and [Serial Number](#) from V-ONE to obtain a License Key and certificate.

Detailed instructions for acquiring a license are included in "Obtaining Your License and Certificate" in both [Chapter 3](#), "Installing the SmartGate Server Software—UNIX" and [Chapter 4](#), "Installing the SmartGate Server Software—Windows NT."



# Chapter 2

## Preinstallation of the SmartGate Server Software

This chapter provides preinstallation information for installing the SmartGate Server software.

### Perform Preinstallation

Table 2–1 is a checklist of the tasks that you must complete in order to set up a SmartGate Server. Details on completing each task follow the checklist.

**Table 2–1**  
*SmartGate Server Setup Checklist*

Steps Required	Page/Chapter
Set up your TCP/IP Protocol properties prior to installation of SmartGate on a Windows platform	<a href="#">Page 30</a>
Obtain your installation values	<a href="#">Page 31</a>
Set up your Oracle SQLNet System (for Oracle users only)	<a href="#">Chapter 9</a>
Set up your ACE/Server (for RSA SecurID users only)	<a href="#">Chapter 6</a> and RSA Security, Inc. documentation
Set up your RADIUS Backend Server (for RADIUS users only)	<a href="#">Chapter 6</a> and RADIUS documentation
Set up your <a href="#">Netrust CA Server</a> (for Netrust users only)	<i>SmartGate With Netrust Authentication Guide</i> and Netrust's own documentation
Set up your <a href="#">Entrust CA Server</a> (for Entrust users only)	<a href="#">Chapter 6</a> and Entrust documentation

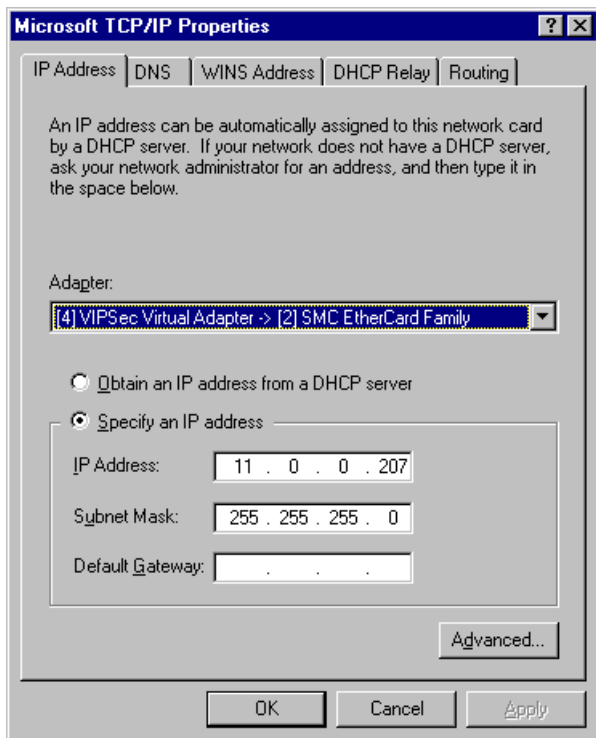
## Setting Up Your TCP/IP Protocol Properties

V-ONE recommends that all network TCP/IP protocol properties be established according to your network configuration on your server prior to installing the SmartGate 4.x Server software.

TCP/IP properties include:

- Specify an IP address and Subnet Mask
- Specify the Default Gateway
- Specify the WINS Servers addresses
- Specify the DNS Servers
- Enable Routing—If IPSec is being used, IP forwarding must be turned on

To access your TCP/IP protocol properties on a Microsoft Windows NT Server, double-click **Network** in the Windows Control Panel. Click the **Protocol** tab, select **TCP/IP**, and click **Properties**. An example of the IP Address—TCP/IP Protocol Properties Window is displayed in Figure 2-1.



**NOTE:** IP addresses cannot be easily changed if the IPSec driver is installed. Make certain the settings are correct in the Windows control panel.

**NOTE:** For detailed instructions on configuring IPSec, see Chapter 11, "IPSec."

**Figure 2-1**  
**TCP/IP Protocol Properties Window**

**NOTE:** Information contained in screen shots are for display purposes only. You should input the information that correctly reflects your network/SmartGate System.

## Obtain Your Installation Values

The following information will be needed either during installation or setup of the SmartGate Server.

### Administrator ID

Your UNIX Account Administrator ID is usually “root.”  
Your Windows NT Account Administrator ID is usually defaulted to “Administrator.”

### Administrator Password

The sequence of characters that, when combined with your Administrator ID, authorizes administrative access to your server’s operating system.

### SmartGate Server’s IP Address(es)

If the server has multiple interfaces, you will need this information for each interface.

### SmartGate Server’s Domain Name

Identifies a ‘location’ on the Internet (e.g., v-one.com) that has been registered with the Internet Network Information Center (InterNIC). Currently the domain name is limited to 47 characters. Through the use of aliases, however, it is possible to accommodate longer names.

### SmartGate Server’s Authenticator

The name assigned to a SmartGate Server through which users can access a particular service. This name can be up to 14 alphanumeric characters in length and it is recommended that it be a derivative of your SmartGate Server hostname.

### SmartGate Administrator’s User ID

Each SmartPass user is assigned a unique string of characters in the SmartGate user database which will identify only that user. This unique string is called a [User ID](#). A User ID is associated with the user’s authentication key. It can be up to 30 characters in length, cannot include spaces or special characters (\*,’,/), and must be unique on the SmartGate Server where it resides. When a user has SmartGate administration privileges, that user’s User ID is known as the SmartGate Administrator’s User ID.

**NOTE:** A User ID defaults to the authentication token manufacturer’s parameters. For example, the MCOS and STARCOS smart cards have a limit of 15 characters for the User ID. Therefore, because the FIPS and VCAT virtual tokens use an emulation of the MCOS smart card, they also only support 15 character User IDs.

## **SmartGate Group**

The identifier that allows you to assign users to organizational groups. The group's name can be up to 23 characters in length and cannot include spaces or special characters (\*,/,).

If a hierarchical organizational structure is desirable, keep in mind that SmartGate allows you to group users together and give one group of users a common set of access permissions. In addition, groups can “inherit” access permissions from other groups in a hierarchical manner. Before you start adding users to the user database, mapping out your group deployment structure is a useful step. Each user (uniquely identified by a “User ID”) has an entry for “group” in the user database. In addition, all users in the SmartGate community are members of the universal group “all.”

## **Customer ID**

The customer's identification number. The Customer ID along with the Serial Number, which also identifies the customer, are used to generate a License Key and certificate. You will receive your Customer ID by e-mail from V-ONE upon registration of the software.

## **Serial Number**

A number used to identify the customer. The Serial Number along with the Customer ID, which also identifies the customer, are used to generate a License Key and certificate. You will receive your Serial Number by e-mail from V-ONE upon registration of the software.

## **RADIUS Backend Server Hostname (RADIUS Only)**

The fully qualified domain name or IP address of the computer that will serve as the RADIUS Backend Server(s). There may be up to five RADIUS Backend Servers.

## **RADIUS Backend Server Shared Secret Code (RADIUS Only)**

The shared secret code between the SmartGate Server running the RADIUS module and the RADIUS Backend Server(s). There may be up to five RADIUS Backend Servers. Each RADIUS Backend Server must be configured with its corresponding shared secret code. See your RADIUS documentation for further information.



### **Entrust/Netrust Directory (Entrust or Netrust Only)**

The directory where your Entrust/Netrust files (specifically `entrust.ini`) were installed. The `entrust.ini` file contains the location of the Entrust or Netrust CA Server and Manager and is used by both the SmartGate Server and SmartPass.

If you are installing the SmartGate Server on a UNIX platform, you will also need the directory where the Entrust run time and session libraries are located.

### **Entrust/Netrust Authorization code (Entrust or Netrust Only)**

The Authorization code issued by the Entrust or Netrust CA Server.

### **Entrust/Netrust Reference Number (Entrust or Netrust Only)**

The Reference number issued by the Entrust or Netrust CA Server.

### **PKI (PKI Authentication Only)**

The Server Certificate and a PKI PKCS #12 file (.p12 or .pfx).



# Chapter 3

# Installing the SmartGate Server Software—UNIX

**NOTE:** Before installing the SmartGate Server software on your system, you must have superuser status, either as the [UNIX administrator](#) or by logging on as “root.”

This chapter provides instructions for installing the SmartGate Server software in a UNIX environment.

## Installing the Software on Your System

Follow the procedures in this section for both new installations and upgrades.

## Upgrading from a Prior Version

The SmartGate installation script will determine if your computer system has a prior version of the SmartGate Server software installed. If it does, the installation script assumes that you are upgrading and will automatically back up the relevant files.

You can either set your SmartGate Server's root directory or use the system default of `/usr/smartgate`.

During installation, the script will rename all the old executable files with an extension of `.sgn` and move them into `/SmartGate Server's root directory/old`. Updated versions of the executable files will then be installed in the appropriate directories. Configuration files, such as `reginfo.dat` and all `.acl` and `.db` files are not overwritten during an upgrade. However, a backup of the user database (`sgusr.db`) and all `.acl`, `.dny`, and `.grp` files will be written to `/SmartGate Server's root directory/old_db`.

**It is highly recommended that a complete backup be performed before installing the new SmartGate Server software.**

## General Installation Instructions

The initial steps of the installation procedure are:

1. Log on as **root**.
2. Mount the CD-ROM. Type:  
**mount<space>-o<space>ro<space>/cdrom**
3. Change your directory into the appropriate operating system. For example, type:  
**cd<space>/cdrom/sgserver/bsdi/3.x-4.x/install**
4. On the UNIX system where your SmartGate Server will reside, use the **mkdir** command to create a temporary directory (e.g., `/usr/sg`). This is the SmartGate installation directory, which will contain the installation files. For example, type:  
**mkdir<space>/usr/sg**
5. Copy the appropriate files from the CD to your temporary directory. For example, type:  
**cp<space>\*<space>/usr/sg**
6. Use the **cd** command to switch to the SmartGate installation directory that you set up as described above. For example, type:  
**cd<space>/usr/sg**
7. You are now ready to run the installation script.

## Running the Installation Script

Before you run the SmartGate installation script, you should use the **script** command to create an installation script file, which will contain a transcript of your session. For example, type:

```
script<space>install.yyyymmdd.log
```

The system responds with:

```
Script started, output file is install.20000201.log
```

For a SmartGate standard installation, run the installation script by typing:

```
./install
```

The SmartGate Server Software Main Menu (Figure 3-1) will be displayed.

**NOTE:** The installation requires at least 15 megabytes of space in the `/tmp` directory as temporary swap space.

**NOTE:** To avoid having to retrieve the installation files repeatedly, do not place them in the `/tmp` directory. Most systems erase the `/tmp` directory when the system is rebooted.

**Figure 3-1**  
**SmartGate Server Software**  
**Main Menu**

**NOTE:** Only the options available at any given time will be visible on each menu.

**NOTE:** In any menu, the letter or number in the brackets “[ ]” is the key you should press followed by ENTER in order to perform that particular action.

**NOTE:** If at a later time you decide to reconfigure the SmartGate Server software, run  
./setup from /SmartGate  
Server's root  
directory/bin.

**Figure 3-2**  
**SmartGate Server Software**  
**Installation Menu**

**NOTE:** If you are upgrading from 2.3 or earlier, you will have the option not to upgrade old SmartGate configuration files.

**NOTE:** None of the commands are case sensitive, but the information entered by the user may be, and often is, case sensitive.

SmartGate Server Software Main Menu

Press

- [I] To install the SmartGate Server software
- [C] To configure SmartGate Server software
- [B] To back up current configuration files
  
- [R] To open the ReadME document
  
- [U] To uninstall the SmartGate Server software
  
- [?] For Help
- [X] To exit program

Command ==>

Press **I** to start the installation process. The License Agreement will be displayed. It must be accepted to proceed with the installation. After accepting the License Agreement, Figure 3-2 will be displayed.

SmartGate Server Software Installation Menu

Press

- [H] To change the SmartGate root directory from /usr/smartgate
- [E] To change the administrators' e-mail address (es)  
Current value is root
  
- [C] To enable/disable overwrite of old SmartGate jobs for  
scheduler daemon (cron) Current value is enable
- [I] To enable/disable removal of nonessential jobs for Internet  
services daemon (inetd) Current value is to disable
- [P] To enable/disable installation of Perl 5 executable  
Current value is disable
- [M] To enable/disable changes to Message of the Day file  
Current value is disable
  
- [S] To start installation
  
- [?] For Help
- [Q] To quit the program
- [X] Return to the Main Menu

Command ==>

1. **[H] Change the SmartGate root directory** The default SmartGate Server's root directory is /usr/smartgate. However, you may designate a different directory.
2. **[E] Change administrators' e-mail address** The default e-mail address for the administrator is set to "root."
3. **[C] Enable/disable of old SmartGate jobs** The default of enable will remove the old SmartGate cron jobs. Only if you are upgrading from a previous version of SmartGate and you had customized your cron tabs, should you set this option to **disable**.

4. **[I] Enable/disable removal of nonessential jobs** Select enable, if you want to remove all nonessential Internet services for security reasons. The services are not actually deleted they are just commented out. The system is defaulted at disable.
5. **[P] Enable/disable installation of Perl 5** The default of disable will not install the Perl 5 executables on your system. If you do not have Perl 5 on your system, the SmartGate log reporting function will not work.
6. **[M] Enable/disable changes to Message of the Day file** If you want SmartGate to write to the Message of the Day file enable this function. The Message of the Day is included for backward compatibility.
7. **[S] Start installation** Once the installation is started by pressing S, Figure 3–3 will be displayed.

```
-----  
Please enter the hostname or IP address of this machine:  
[ambrosia.fence.v-one.com]  
Please enter the Authenticator name of this machine: [ambrosia]  
Please enter the Inside IP of this machine: [10.0.0.225]  
The SmartGate Server software will now be installed on your  
machine. If for any reason the installation was not successful,  
please contact 1-888-220-VONE (8663) or support@v-one.com.  
Press Enter to continue.  
SmartGate must remove all spaces from your Group Names. However,  
before doing this a backup of your user database, ACLs, and group  
files will be created and placed into /SmartGate Server's root  
directory/old_db.  
Press Enter to continue.  
-----
```

You are prompted for the three required configuration parameters. The installation script will attempt to obtain the appropriate information and insert it as a default value in the brackets [ ]. You can change these values as needed.

1. **Hostname** sets the `domainname` value, the hostname or IP address of your SmartGate Server (specifically, where your OLR Server resides). For licensing reasons, please make certain that your hostname is unique. Do not use a generic name like “SGServer.”
2. **Authenticator name** sets the `authenticator` value. This setting is used by SmartPass to associate secure paths with this SmartGate Server. It can be any string of up to 14 alpha-

**Figure 3–3**  
**SmartGate Server Software**  
**Install Prompt Menu**

**NOTE:** Not all text is being displayed in this figure.

**WARNING!** For licensing reasons, please make certain that your hostname is unique. Do not use a generic name like “SGServer.”

**NOTE:** SmartGate uses the program `dbconv` to remove spaces from all SmartGate group names, however this program will only run during upgrades—not new installations.

**NOTE:** The hostname should be an FQDN or IP address EXCEPT if you are using UDP broadcasting (UDPPortList) than the IP address MUST be used.

**WARNING!** For licensing, your hostname is case specific. For example, capital and lowercase letters can be used, but they must match exactly to the hostname in the license.

numeric characters but it is recommended that it be based on or be a derivative of your SmartGate Server's hostname.

3. **InsideIP** sets the [InsideIP](#) value. It enables you to specify the IP address assigned to the inside network adapter card on your SmartGate Server.

These values are displayed in SmartAdmin under "[System Definition Settings](#)," in Chapter 5, "Using SmartAdmin."

If you are upgrading from a previous version of SmartGate, the program, `dbconv`, will automatically search your user database, access control lists (ACLs), and group files and remove all spaces from SmartGate group names if necessary. A backup will automatically be made of all the pertinent files and stored in `/SmartGate Server's root directory/old_db`.

## Obtaining Your License and Certificate

Downloading your License Key and certificate from the V-ONE Web site is the final necessary step to getting your SmartGate Server up and running.

The installation script will display the following brief directions on how to obtain your License Key and certificate (Figure 3-4).

**Figure 3-4**  
**SmartGate Server Software**  
**License Instructions Menu**

**NOTE:** If your server is not connected to the Web and you are downloading the SmartGate software onto another computer, use the FTP or Telnet function to transfer files from one machine to another.  
See "How To FTP" in Appendix B, "Services" for instructions on how to FTP a file.

```
To complete your installation, you need to open a Web browser and
navigate to the Get SmartGate download site:

    http://license.v-one.com/license/html/gtsmartgate.html

On the Get SmartGate Web site, click the REGISTER button and
enter the appropriate information to register your SmartGate
Server software. You will be e-mailed a Customer ID and Serial
Number.

Return to the Get SmartGate Web site and click the GENERATE
LICENSE KEY button. Follow the instructions to upload the
keyname.has, keyname.pub, and sgc.ini files to V-ONE and then
download the License Key (vone.lic) and certificate
(keyname.cer). The License Key should be copied to /SmartGate
Server's root directory/etc/keys and the certificate to
/SmartGate Server's root directory/etc.

You must reboot after installing the SmartGate Server software,
License Key, and V-ONE certificate.

Press Enter to continue.
```

Detailed instructions are given below:

1. Register your SmartGate software:

- Open a Web browser and navigate to:  
<http://license.v-one.com/license/html/gtsmartgate.html>
- Click **Register**, type in the requested information, and click **Submit**.
- V-ONE will send a Customer ID and Serial Number to the e-mail address specified during registration.

2. Generate and install your License Key:

- Open a Web browser and navigate to:  
<http://license.v-one.com/license/html/gtsmartgate.html>
- Click **Generate License Key** and type in your Customer ID and Serial Number.
- Upload to V-ONE the *keyname.has*, *keyname.pub*, and *sgc.ini* files by using the browse buttons to select the location of each file.
- After uploading the *keyname.has*, *keyname.pub*, and *sgc.ini* files, the Web page will display hyperlinks to download *vone.lic*, which is your License Key, and *keyname.cer*, which is your certificate, directly from the V-ONE Web site to the appropriate location.
- Download the *keyname.cer* file to /SmartGate Server's root directory/etc.
- Download the *vone.lic* file to /SmartGate Server's root directory/etc/keys.

3. The SmartGate Server must be rebooted after installation of the server software, however, the server can be configured prior to reboot.

After installing the SmartGate Server software and the License Key and certificate files, all further configuration can either be performed through the UNIX console `setup` script or by using SmartAdmin, SmartGate's administration GUI. In order to use SmartAdmin remotely you will need to perform the following minimal configuration.

**NOTE:** V-ONE requires that you have your *keyname.has*, *keyname.pub*, and *sgc.ini* files ready to upload to V-ONE at this time. All of these files reside in /SmartGate Server's root directory/etc.

**NOTE:** You must use the browse button, you cannot type in the location of the file. If your Web browser software does not support the upload functionality and the browse buttons do not appear, you will need to upgrade your browser.

**NOTE:** If your server is not connected to the Web and you are downloading the SmartGate software onto another computer, use the FTP or Telnet function to transfer files from one machine to another.

See "How To FTP" in Appendix C, "Services" for instructions on how to FTP a file.

**NOTE:** To display your complete license information at any time, run `./vldadmin` from /SmartGate Server's root directory/bin on a UNIX console.



**NOTE:** After installation of the SmartGate Server software, the server must be rebooted.

**NOTE:** If you are using a UID Server to assign User IDs, see “[UID Server for On-Line Registration](#)” in Chapter 7, “On-Line Registration Services” for more information.

**NOTE:** Check that the smart card is in the reader (if a physical smart card is being used).

**NOTE:** You must have UNIX root superuser privilege to perform console administration.

## Minimal UNIX Configuration for Remote Administration

In order to perform any remote administration, such as SmartAdmin, a user must have administrative privileges. In order to obtain the necessary privileges to be a SmartGate administrator, the following steps must be performed:

1. Install the SmartPass software on your personal computer. See the *SmartPass Administrator's Guide* for installation instructions on your operating system.
2. Perform On-Line Registration (a default OLR template has already been installed). See the *SmartPass Administrator's Guide* for more information.
3. Open the SmartPass User Interface to display your User ID.
4. Enable your User ID. At the SmartGate Server console, from /SmartGate Server's root directory/bin, type the following command:

**`./sgadm -enable user`**

where *user* is your User ID.

5. Reenter the UNIX software setup menu. From /SmartGate Server's root directory/bin, type:  
**`./setup`**
6. Give yourself administrative privileges. See “[Adding Administrative Privileges \(adm-gw.acl\)](#)” later in this chapter.

Install and use SmartAdmin for all further configuration. See [Chapter 5, “Using SmartAdmin”](#) for detailed instructions.

Minimal configuration of the SmartGate Server is complete. If you have set yourself up as a remote administrator, any further configuration can be done using SmartAdmin, SmartGate's administration GUI, from a remote Microsoft Windows machine running SmartPass.

## Setting Up the SmartGate Server at the UNIX Console

To continue configuring the SmartGate Server software from the UNIX console run `./setup` in the `/SmartGate Server`'s root directory/`bin`, press **C** in the Main Menu. Figure 3-5 will be displayed.

```
SmartGate Server Software Setup Menu

Press
[K] To regenerate a public/private key pair
[L] Licensing

[C] To configure the SmartGate Server software (sgconf.ini)
[R] To set up On-Line Registration
[A] To set up access permissions
[J] To view SmartGate Server single port proxy configuration
[E] To configure SmartGate extensible components

[?] For Help
[Q] To quit the program
[X] Return to the Main Menu

Command ==>
```

This menu allows you to configure or view all of the SmartGate Server software files. All of the functions of the SmartGate Server Software Setup Menu are also available using SmartAdmin.

## Regenerating the SmartGate Public/Private Key Pair

The SmartGate **public/private key** pair system is the heart of the data encryption performed by SmartGate. This menu provides the means to regenerate your public/private key pair. Press **K** in the Setup Menu and Figure 3-6 will be displayed.

```
Key Regeneration and Testing Menu

Press
[S] To change the key size from: 768

[G] To generate the public/private key pair

[T] To test the public/private key pair
[C] To test the certified key

[?] For Help
[Q] To quit the program
[X] Return to the Setup Menu

Command ==>
```

**Figure 3-5**  
*SmartGate Server Software Setup Menu*

**Figure 3-6**  
*Key Regeneration and Testing Menu*

**NOTE:** Your public/private key pair was automatically generated during installation. If you generate a new public/private key pair, they must be forwarded to V-ONE for certification.

**Figure 3-7**  
**Public/Private Key Pair**  
**Generation Window**

The encryption keyname will be randomly generated by the system and then used to identify your certification files. Use the Key Generation and Testing Menu to perform the following steps:

1. **[S] Change key size** The default key size is 768. This is a standard size for optimum security and performance, however the number may be changed to one of three sizes. For example:

Enter the size of the key pair (512, 768, 1024) :

2. **[G] Generate public/private key pair** A public/private key pair is automatically generated during installation. You do not need to generate a new one. If, however, you want to generate a new key pair, press **G**. Figure 3-7 will be displayed.

```
Type at least 20 random key strokes :
Please wait while generating key pair...
Key done. Save to file keyname.pub and keyname.prv
password encrypt/decrypt check OK
MD5 hash value for public key is:
D71F0E2246EF99476596C09C11FD9221
Public key MD5 Hash value is written to keyname.has
```

Type in key generation material, a random combination of over 20 letters and numbers. You do not have to record your entry. The system will use this entry, along with your encryption keyname, to generate the public/private key pair that will be assigned to your On-Line Registration system. The results will be placed in your certification files, *keyname.pub* and *keyname.prv*.

3. **[T] Test the public/private key pair** You may want to test your public/private key pair before sending it to V-ONE. This function is optional.
4. **[C] Test the certified key** After receiving your certificate from V-ONE, copy the files into /SmartGate Server's root directory/etc. You may want to test your certificate after receiving it from V-ONE. This function is optional.

## Adding a License or Viewing Your License Key

To display your License Key, press **L** in the Setup Menu. Figure 3-8 will be displayed.

```
SmartGate License
Hard Limit : 100
Soft Limit : 90

[A] Add a license:
[D] Delete a license:

Features :

[?] For Help
[Q] To quit the program
[X] To return to the Setup Menu
```

**Figure 3-8**  
**License Key Manager Menu**

**NOTE:** The total hard limit is the license maximum number of users and the soft limit is the point at which the administrator will be notified that the user license limit is being reached. This information is written to the `/var/log/smartgate` file.

**NOTE:** PKI authentication requires License code 54.

## Configuring the SmartGate Server Software (`sgconf.ini`)

Configure the `sgconf.ini` file, by pressing **C** in the Setup Menu. Figure 3-9 is displayed.

```
SmartGate Server Software Configuration Menu (sgconf.ini)

Press
[1] To change AccessCodeDaysValid from : 0
[2] To change accounting_service from :
[3] To change authenticator from : ambrosia
[4] To change anon_reg_allowed from :
[5] To change AuthEncryptMethod from : DES
[6] To change AuthMethod from :
[7] To change backup_userdb from :
[8] To change debug from : 0
[9] To change denial_server from :

[N] Go to the next set of values

[?] For Help
[Q] To quit the program
[X] Return to the Setup Menu

Command ==>
```

**Figure 3-9**  
**SmartGate Server Software  
Configuration Menu  
(`sgconf.ini`)**

**NOTE:** Not all options are being displayed. For detailed descriptions of each `sgconf.ini` option, either refer to [Appendix A](#) or press the number of the option and a description will be displayed with the prompt.

The Configuration Menu consists of multiple screens with up to 9 options on each screen. Type the line number of the option you want to modify and press **ENTER**. The installation script describes the option and allows you to enter or change the necessary information. Changes to the configuration become effective immediately. It is not necessary to reboot the system. There are only three values that must be set in order for SmartGate to run. These values should have been set during the installation procedure.

1. Hostname or IP address
2. Authenticator
3. Inside IP

During a first-time installation, the defaults will be in effect. However, if this is an upgrade from a previous version, the original `sgconf.ini` options will be retained and this menu will reflect the previous configuration.

## Setting Up the On-Line Registration File (`reginfo.dat`)

The next step in configuring your SmartGate Server software is setting up your registration file for On-Line Registration (OLR). This file defines the data entry fields displayed to end users when they perform OLR. Different OLR methods, such as Entrust, may be defined using a section header followed by separate sections of data entry fields. This file is preformatted with the first two default informational fields. Your OLR files can be created here or through SmartAdmin.

Press **R** in the Setup Menu if you want to set up your OLR files now. The SmartGate On-Line Registration Setup Menu (Figure 3-10) will be displayed.

**Figure 3-10**  
**SmartGate On-Line Registration**  
**Setup Menu**

```
SmartGate On-Line Registration Setup Menu

Press
[T] To set up user information (reginfo.dat)
[D] To set up Web page branding (sgconf.ini)

[?] For Help
[Q] To quit the program
[X] Return to the Setup Menu

Command ==>
```

Press **T** to enter the User Information Setup Menu (`reginfo.dat`). An example of a configured menu is displayed in Figure 3-11.

```

User Information Setup Menu (reginfo.dat)
[V] ONE      NE [T] RUST      ENTRU [S] T      [P] KI
OLR method: VONE

Informational Fields are
1.  First Name:20:alphanum
2.  Last Name:20:alphanum
3.  Phone No.:20:phone
4.  Soc.Sec.No.:11:ssn
5.  Credit Card No.:20:ccnumber
6.  Exp. Date (mmyyyy):6:numeric
7.  Street Address:60:alphanum
8.  City/State/Zip:60:anything
9.  Group:20:grouplist:finance;marketing;sales
10. E-mail Address:40:anything

Press
[P] To move a field up
[N] To move a field down
[E] To edit a field
[A] To add a new field
[D] To delete a field

[?] For Help
[Q] To quit the program
[X] Return to the OLR Setup Menu

Command ==>

```

**Figure 3-11**  
**User Information Setup Menu**  
(reginfo.dat)

**NOTE:** The first two fields (in bold) are defaults. The field names and sizes can be changed, but not the field types.

**NOTE:** Netrust and Entrust are available only on Solaris and the Microsoft Windows NT operating systems.

V-ONE is the default OLR method. Press **T** to configure the data entry fields for Netrust authentication, **S** for Entrust authentication, or **P** for PKI authentication. The NETRUST and ENTRUST options will only be visible on this menu if they are enabled in the Extensible Components Menu.

This menu may contain up to 10 lines, one for each data entry field that may be requested of the user. The first two informational fields are filled in with default values. Those fields are required, but the field names and sizes can be changed.

During a first-time installation only the first two fields will be listed. However, if this is an upgrade from a previous version, the original reginfo.dat information will be retained and this menu will reflect the previous setup.

The options below will allow you to adjust reginfo.dat accordingly:

1. **[P] Move field up/[N] Move field down** Use these features to rearrange the order of the fields.
2. **[E] Edit a field** To edit an existing field, press **E**. You will be prompted to enter the line number of the field you want to edit and then the Add/Change Registration Fields Window (Figure 3-12) will be displayed.

**Figure 3-12**  
**Add/Change Registration Fields**  
**Window**

**WARNING!** The field type for fields 1 and 2 must be alphanumeric.

**NOTE:** Fields 3 through 10 are not required by the system. However, when configured, entry is mandatory. The end user will be unable to continue with OLR if information is not entered into each information field.

**NOTE:** `passnum` and `password` are forced confirmation field types. For each field of these types that are requested, a confirmation dialog box will be displayed after the user presses **Next** during OLR.

3. **[A] Add a new field** Press **A** to add a new field. You will be prompted for a new field name, size, and type, and then Figure 3-12 will be displayed with this new information.

Add/Change Registration Fields

Press

[F] To change field name from: Soc. Sec. Number

[S] To change field size from: 11

[T] To change field type from: ssn

[?] To Help

[Q] To quit without saving changes

[X] To make change permanent

Command ==>

- **Fields 1 and 2 (Required)** Field 1 and field 2 together make up the user's long name in the user database. They are labeled, "First Name" and "Last Name" by default, but they can be changed. The field names should be easy for your end users to understand. If you want to capture the full name of your user, retain the default values. The user's long name is the first field in `reginfo.dat` followed by the second field, with a blank in between. For example:

```
users_long_name=John Smith
```

- **Fields 3 to 10** You can define any additional information you want to obtain from each end user.

The following list describes the different data types that can be assigned to each field.

Type	Valid Entries
alphanumeric	alphabetic, 0-9
numeric	0-9
phone	0-9, comma (,) dash/hyphen (-)
grouplist	alphabetic, 0-9
ccnumber	0-9
passnum	0-9
password	alphabetic, 0-9
ssn	0-9, dash/hyphen (-)
anything	no type check

If the type "**grouplist**" is used, the group names, which will appear in a drop-down list during OLR, must be entered as a semicolon delineated list. For example,

```
grouplist:finance;marketing;sales
```

4. **[D] Delete a field** Use this feature to delete an existing field.

When your users connect to the OLR Web page (using SmartPass), it will reflect the information fields defined in `reginfo.dat`.

## Branding the On-Line Registration Web Page (`sgconf.ini`)

Using Web page branding, your OLR Web confirmation page can be configured to reflect your company information. You may also create a desktop shortcut for your end users which will launch SmartPass and a Web browser to a specific Web page. Press **R** in the Setup Menu, and then press **D** in the SmartGate On-Line Registration Setup Menu. Figure 3–13 will be displayed.

Web Page Branding Menu (`sgconf.ini`)

```
[I] To change company name from :
[A] To change street address from :
[C] To change city name from :
[S] To change state/province name from :
[Z] To change zip code/postal code from :
[R] To change country code from :
[P] To change phone number from :
[E] To change e-mail address from :
[U] To change company's Web page address from :

[D] To change shortcut description from :
[W] To change default Web page from :
[F] To change whether all users are outside firewall from:

[?] For Help
[Q] To quit the program
[X] Return to the OLR Setup Menu

Command ==>
```

**Figure 3–13**  
**Web Page Branding Menu**  
(`sgconf.ini`)

**NOTE:** When changing the OLR branding on the SmartGate Server, an administrator may not enter an “\*” because of wildcard matching.

1. **[I] [A] [C] [S] [Z] [R] [P] [E] [U] Company Information** Use these options to brand the OLR Web page with your company's name and address.
2. **[D] Shortcut description** Enter the title you want to appear under the shortcut SmartPass icon that will be placed on your end users' desktop after they register.
3. **[W] Default Web page** Enter the [URL](#) of the Web page, such as your company's home page, that will be launched along with SmartPass using the shortcut SmartPass icon.
4. **[F] Users' outside firewall** Set this option to **yes** if all of your users have direct access to the Web without passing through a firewall.



## Configuring Access Permission Files (sgate.acl and sweb.acl)

Another step in configuring your SmartGate Server software is to set up your access permissions environment. The idea is to create a basic set of groups and to assign each a set of permissions. As administrator, you will want to map a logical set of groups at the beginning of this process.

There are multiple ways to administer permissions after the initial configuration process is finished. You can either run `./setup` in `/SmartGate Server's root directory/bin`, or you can use SmartAdmin for remote administration. See “TCP and Web Access Permissions” in Chapter 5, “Using SmartAdmin” for more information.

In order to set up your access permissions using the installation script, press **A** in the Setup Menu. Figure 3–14 will be displayed.

**Figure 3–14**  
**Access Permissions Menu**

```
Access Permissions Menu

[U] To change user access permissions (sgate.acl & sweb.acl)
[A] To change administrator access permissions (adm-gw.acl)

[?] For Help
[Q] To quit the program
[X] To return to the Setup Menu

Command ==>
```

Press **U** to configure access permissions for individual users and groups. Figure 3–15 is displayed.

**Figure 3–15**  
**User Access Permissions Setup**  
**Menu (sgate.acl & sweb.acl)**

```
User Access Permissions Setup (sgate.acl & sweb.acl)

To change Secure Path Type, press
[H] TTP [F] TP [T] ELNET [P] OP-3
[S] MTP [O] RACLE [W] NNTP [O] THE [R]

Secure Paths for HTTP:
1. ~all /website:8000/
2. ~all /website:8001/
3. ~all /website:12500/
4. ~all /website.europe.global.v-one.com:8000/
5. ~all /website.europe.global.v-one.com:8001/

[N] Go to the next page of secure paths
[A] To add a HTTP secure path [E] To edit a HTTP access permission
[D] To delete a HTTP secure path [U] To undo changes

[?] For Help
[Q] To quit the program
[X] To save and exit to the Setup Menu

Command ==>
```

1. **[H] [F] [T] [P] [S] [O] [W] [R] Change secure path type**  
Enables you to designate the secure HTTP, FTP, TELNET, POP3, SMTP, ORACLE, or NNTP connection(s) that the user can access via SmartGate.

2. **[A] Add secure path** Use this function to add secure paths to the secure path type you are working with (HTTP, FTP, TELNET, ...). For example, if you are adding a secure Web access pathway (HTTP) you would see these prompts:

```
Add HTTP service to what group/user?  
(please use a ~ to denote a group name) ~all  
Add HTTP service to what destination host (s) ?  
(please use a ~ to denote a group name)  
eden.fence.v-one.com  
Enter service port on destination [80] :  
Enter service port on SmartGate Server [2080] :
```

3. **[E] Edit secure path** Use this function to edit an existing secure path. For example, if you are editing a secure Web access pathway (HTTP) you would see these prompts:

```
Which HTTP access permission do you want to edit ?  
Change owner of HTTP permission from ~all to:  
[all] :  
Change destination of this access permission  
[eden.fence.v-one.com] :  
Change the destination port of this access permission  
[80] :  
Change the server port of this access permission  
[2080] :
```

4. **[D] Delete secure path** Use this function to delete an existing secure path. For example, if you are deleting a secure Web access pathway (HTTP) you would see these prompts:

```
Enter path number to be deleted:  
Are you sure you want to delete HTTP service to  
/www.v-one.com/? [yes] /no
```

**NOTE:** The value in the brackets [ ] is the default for that type of secure path (HTTP, FTP,...) and should not be changed.

**NOTE:** As long as you are working directly at the SmartGate Server and have superuser status or are logged in as `root`, you are considered a Superuser Administrator with access to any group.

**NOTE:** If you want to use the UID Server to manually assign User IDs for yourself and other users, see “UID Server for On-Line Registration” in Chapter 7, “On-Line Registration Services.”

**Figure 3-16**  
**Administrator Privileges Setup**  
**Menu (adm-gw.acl)**

**NOTE:** When this menu is opened for the first time, no administrators will be assigned.

## Adding Administrative Privileges (adm-gw.acl)

In order to perform remote administration a user must be assigned administrative privileges. In order to assign yourself (or another user) superuser privileges for remote administration, you must configure the `adm-gw.acl` file, located in the SmartGate Server’s root directory.

- Add your User ID to `adm-gw.acl`.

In order to add a User ID to `adm-gw.acl`, the User ID must be generated. If you want to have SmartGate generate your User ID through OLR, you must first exit the installation script, reboot the SmartGate Server, and perform OLR remotely. However, you will not be able to perform OLR until you have received your certificate from V-ONE. The installation script can be reentered by running `./setup` in /SmartGate Server’s root directory/bin.

- Assign an administrative level (`superuser`, `standard`, `restricted`, or `minimal`).
- The groups you are allowed to administer must be assigned.

Press **A** in the Setup Menu to open the Access Permissions Menu. To give administrative privileges to a user, press **A** in the Access Permissions Menu. Figure 3-16 will be displayed.

Administrator Privileges Setup Menu (adm-gw.acl)

Superuser Administrators  
User Patt52598 for group(s) any  
User jsmith for group(s) any

Standard Administrators  
User Shar15838 for group(s) management

Restricted Administrators  
User Joez26359 for group(s) any

Minimal Administrators  
User Robe36352 for group(s) any

[A] To add an administrator  
[D] To delete an administrator  
[E] To edit an existing administrator  
  
[?] For Help  
[Q] To quit the program  
[X] To return to the Access Permissions Menu

Command==>

The Administrator Privileges Setup Menu is an example of various levels of administrative privileges. There are four different levels that can be assigned to a Remote Administrator:

- **Superuser Administrator(s)** have full access to all settings. In addition to the privileges of the standard administrator, superusers can assign administrator levels, change SmartGate configuration settings, and configure the OLR, IPsec Channels, PKI, and Single Port Proxy Map files. Superusers have full access to all groups.
- **Standard Administrator(s)** have full access to all access permissions. In addition to those rights provided at the restricted level, administrators can add/edit/delete access permissions. Access at this level may be limited to certain groups.
- **Restricted Administrator(s)** have full access to user data. In addition to those rights provided at the minimal level, administrators can change authentication keys and add/edit/delete end users. Access at this level may be limited to certain groups.
- **Minimal Administrator(s)** can only enable/disable users and edit a user's name in the event of a name change or a typographical error. Access at this level may be limited to certain groups.

An administrator can be assigned privileges to one or more groups or they can be assigned to **any**, in which case they can administer all users.

# Configuring Single Port Proxy Services

The SmartGate Single Port Proxy provides the various SmartGate services with a single-port presence on the perimeter of a network. This means that all SmartPass-to-SmartGate (client-to-server) connectivity will pass through the Single Port Proxy and be forwarded to the correct destination SmartGate service.

The Single Port Proxy must determine the SmartGate service destination for each SmartPass connection arriving on its single port. To do this, it uses a port mapping file, `sgproxy.conf`. A default rules file is installed with the SmartGate Server software. Use SmartAdmin to edit the rules file (Figure 3-17) if necessary.

**NOTE:** The `sgproxy.conf` file is located in the SmartGate Server's root directory.

**Figure 3-17**  
**Single Port Proxy Menu**

**NOTE:** The default for the Single Port Proxy is 3845. If you want to change the default, see “[Changing the Default Single Port Proxy](#)” in Chapter 5, “Using SmartAdmin.”

Single Port Proxy Menu				
Proxy Rules:				
Name	Type	Host	Port	Access Control
SPSGFTP	TCP	127.0.0.1	2022	yes
SGATE	TCP	127.0.0.1	2023	yes
SWEB	HTTP	127.0.0.1	2080	yes
SWEB	HTTP	127.0.0.1	3900	yes
SGORA	TCP	127.0.0.1	3521	yes
OLR	HTTP	127.0.0.1	2090	no
SGREG	HTTP	127.0.0.1	2090	no
SGSDI	HTTP	127.0.0.1	2095	no
SGSDIP	HTTP	127.0.0.1	2095	no
SGENT	HTTP	127.0.0.1	2096	no
SGRAD	HTTP	127.0.0.1	2097	no
SGRADP	HTTP	127.0.0.1	2097	no
SMARTTIME	TIME	127.0.0.1	3845	no
SGTIME	TIME	127.0.0.1	3845	no
SGCCAG	HTTP	127.0.0.1	3844	no
KRAKIT	HTTP	127.0.0.1	3484	no
[?] For Help				
[Q] To quit the program				
[X] To return to the Setup Menu				
Command==>				

The Single Port Proxy is, by default, the preferred method of connectivity for the SmartGate Server and its clients.

## Configuring SmartGate Extensible Components

There are several add-on components available for use with the SmartGate software, including:

- RSA SecurID Authentication
- RADIUS Authentication
- Entrust/Netrust Authentication

Press **E** in the Setup Menu to configure any of these components. Figure 3-18 will be displayed.

```
SmartGate Extensible Components Menu

[D] To enable/disable RSA SecurID authentication services
    at system startup      Current value is disable
[F] To configure the SmartGate/SecurID Server

[R] To enable/disable RADIUS authentication services
    at system startup      Current value is disable
[G] To configure the SmartGate/RADIUS Server

[E] To enable/disable Entrust/Netrust authentication
    services at system startup  Current value is disable
[N] To configure the SmartGate Entrust/Netrust Server

[K] To enable/disable PKI authentication services
    at system startup      Current value is disable

[?] For Help
[Q] To quit the program
[X] To return to the Setup Menu

Command==>
```

**NOTE:** Entrust and Netrust are available only on Microsoft Windows NT and Solaris operating systems.

**Figure 3-18**  
**SmartGate Extensible Components Menu**

**NOTE:** PKI authentication is only available for Solaris.

### RSA SecurID Authentication

RSA SecurID authentication is a two-factor authentication method using the RSA SecurID token and ACE/Server authentication products developed by RSA Security, Inc. SmartGate supports all types of RSA SecurID authentication tokens, including the standard card/key fob, PINPAD card, and SoftID card. The token's microprocessor and host computer are synchronized by a unique number and the time of day. When users log onto a RSA SecurID-enabled host, they are required to type in their Username and passcode, which is a combination of their assigned pincode and the constantly changing number displayed on the token.

RSA SecurID authentication services are disabled on startup. However, if you are using this authentication method, you enable it by pressing **D**.

**NOTE:** Detailed configuration instructions for RSA SecurID authentication are presented in “[Using RSA SecurID for User Authentication](#)” in Chapter 6, “User Authentication.”

**Figure 3-19**  
**RSA SecurID Authentication**  
**Configuration Menu**

**NOTE:** If you are using RADIUS and have 3 or more interfaces, the `InsideIP` must be the interface which connects to the RADIUS Backend Server

**NOTE:** Detailed configuration instructions for RADIUS authentication are presented in “[Using RADIUS for User Authentication](#)” in Chapter 6, “User Authentication.”

There are two settings in `sgconf.ini` that affect RSA SecurID authentication. Press **F** in the Extensible Components Menu and Figure 3-19 is displayed.

```
SecurID Authentication Configuration Menu

[T] To change timeout value from :
[L] To change time to live value from :

[?] For Help
[Q] To quit the program
[X] To return to the Extensible Components Menu

Command==>
```

- **[T] Challenge timeout** Enter the number of minutes that a **Next Tokencode** or **New Pin Code** dialog box will remain on the screen before it times out, or use the default of 3 minutes. The valid range is 1 to 30 minutes.
- **[L] Time to live value** Enter the number of minutes for which a RSA SecurID authentication will remain valid before another authentication is required, or use the default of 30 minutes. The valid range is 1 to 1440 minutes.

## RADIUS Authentication

RADIUS authentication is an open-standard (RFC 2138) authentication protocol and is transported over UDP, not TCP. RADIUS authentication offers secure, easily-passable communication between the client, using the SmartPass software, the SmartGate Server running the RADIUS module, and the RADIUS Backend Server. When users log onto a RADIUS-enabled host, they are required to type in an administrator-provided User ID and password.

RADIUS authentication services are disabled on startup. However, if you are using this authentication method, you enable it by pressing **R** in the Extensible Components Menu.

There are various settings in `sgconf.ini` that affect RADIUS authentication. Press **G** in the Extensible Components Menu and Figure 3-20 is displayed.

```

RADIUS Authentication Configuration Menu

[1] Primary Host : 10.0.0.225
    Primary Host Shared Secret : kjhjks
    Primary Host use CHAP : no           Primary Host timeout : 10

[2] Secondary Host :
    Secondary Host Shared Secret :
    Secondary Host use CHAP :           Secondary Host timeout :

[3] Additional Host :
    Additional Host Shared Secret :
    Additional Host use CHAP :           Additional Host timeout :

[4] Additional Host :
    Additional Host Shared Secret :
    Additional Host use CHAP :           Additional Host timeout :

[5] Additional Host :
    Additional Host Shared Secret :
    Additional Host use CHAP :           Additional Host timeout :

[S] Session Length :           [T] Challenge Time Out :

[?] For Help      [Q] To quit the program
[X] To return to the Extensible Components Menu

Command ==>

```

**Figure 3–20**  
**RADIUS Authentication**  
**Configuration Menu**

Select 1 through 5 to assign the primary, secondary, and additional backup hosts to be used as RADIUS Backend Servers.

- **Primary (Secondary or Additional) Host:** Identify the IP address or hostname of your primary RADIUS Backend Server and any additional backup RADIUS Servers.
- **Primary (Secondary or Additional) Host Shared Secret:** Assign a shared secret code to each RADIUS Backend Server. The shared secret code must also be configured on each RADIUS Backend Server. Please refer to your RADIUS documentation for further information.
- **Primary (Secondary or Additional) Host use CHAP:** Type either **yes** to identify that particular RADIUS Backend Server as using CHAP authentication, or use the default, **no**. The SmartGate/RADIUS Server will then simulate a CHAP exchange and send a CHAP-Password value instead of the normally hashed user-Password attribute.
- **Primary (Secondary or Additional) Host timeout:** Enter the number of seconds SmartGate will wait for a response from each host before it times out, or use the default of 120 seconds. The maximum number is 32767 seconds (do not use commas).



**NOTE:** Entrust and Netrust authentication are available only on Solaris and the Microsoft Windows NT operating systems.

**Figure 3-21**  
**Entrust/Netrust Authentication**  
**Configuration Menu**

**NOTE:** See “Using Entrust for User Authentication” in Chapter 6, “User Authentication,” for more information on Entrust authentication.

**NOTE:** The first three options must be completed for Entrust or Netrust authentication to function.

The last two options are applicable to all hosts being used.

- **[S] Session Length:** Enter the number of minutes the RADIUS authentication will remain valid before another authentication is required, or use the default of 30 minutes. The valid range is 1 to 1440 minutes. Do not use commas.
- **[T] Challenge Time Out:** Enter the number of minutes that a RADIUS challenge dialog box will remain on the screen before it times out, or use the default of 5 minutes. The valid range is 1 to 30 minutes.

### Entrust/Netrust Authentication

The Entrust and Netrust authentication methods allow SmartPass clients to use either an Entrust soft token or a Netrust ready smart card and smart card reader instead of other V-ONE tokens. Both the SmartGate Entrust/Netrust Server and SmartPass obtain their credentials from the Entrust or the Netrust CA Server.

Press **N** to enter the Entrust/Netrust Configuration Menu (Figure 3-21).

```
Entrust/Netrust Configuration Menu

[L] To change the run time library path from :
[I] To change the location of the entrust.ini file from :

[R] To register the SmartGate Server with the
    Entrust/Netrust Authority

[E] To change location of the User ID Server for Entrust from :
[N] To change location of the User ID Server for Netrust from :

[?] For Help
[Q] To quit the program
[X] To return to the Extensible Components Menu

Command==>
```

- Press **L** and enter the directory where the run time libraries ( .so files on a Solaris operating system) are located. These files are obtained from Entrust or Netrust—not V-ONE. The run time libraries are required.
- Press **I** and enter the path name for the `entrust.ini` file.
- Press **R** to register your SmartGate Server with the Entrust/Netrust Authority. You will be prompted to enter the Entrust/Netrust Reference number and Authorization code obtained from your Entrust or Netrust software.

- Press **E** or **N** to specify either the hostname or IP address and the port number of the optional SmartGate UID Server for an Entrust or Netrust UID Server. This option configures the `uid_server` setting in `sgconf.ini`.

Please refer to the *SmartGate With Netrust Authentication Guide* for further information on how to configure your SmartGate System for Netrust authentication.

## PKI Authentication

Public-key infrastructure (PKI) is the combination of software, encryption technologies, and services that enable businesses to protect the security of their communications and business transactions via the Internet.

PKI integrates digital certificates, public-key cryptography, and certificate authorities into enterprise-wide network security architecture. SmartGate/PKI encompasses the verification of digital certificates to individual users and servers, OLR for end-user enrollment, and SmartAdmin for managing certificates.

The configuration setting in `sgconf.ini` that reflects the verification level of the PKI certificates is `TrustedCAList`.

If this value is set to enable and you have added CA certificates (using the `certmanager` program) to the trusted CA list, then the SmartGate PKI authentication server will check:

- the validity of the date of the user certificate,
- verify that the certificate is the same as the original OLR PKI certificate, and
- verify the user certificate is trusted by one of the trusted CAs in the list.

If the value is set to disable then only the date of the user certificate is checked. The validity date check is to verify the “not before date” and the “not after date.”

**NOTE:** The command to reboot your computer is “reboot.”

## Rebooting Your Computer

At this point, you have configured your SmartGate Server software for all basic functions. You will need to reboot your computer before SmartGate will function. Remember to copy your V-ONE certificate (.cer file) to /SmartGate Server's home directory/etc. and your License Key (the vone.lic file) to /SmartGate Server's home directory/etc/keys. You cannot perform OLR until your License Key and certificate have been installed.

## Backing Up SmartGate Configuration Files

You can use the installation script to make a backup copy of selected files, such as sgconf.ini, before configuring them. Press **B** in the Main Menu to open the SmartGate Configuration Files Backup Menu (Figure 3-22).

**Figure 3-22**  
**SmartGate Configuration Files Backup Menu**

```
SmartGate Configuration Files Backup Menu

Press
[T] To change target directory or device from /dev/fd0

File List: /usr/local/etc/sweb.acl /usr/local/etc/sgate.acl
/usr/local/etc/adm-gw.acl /usr/local/etc/sgconf.ini /usr/
local/etc/reginfo.dat /usr/local/etc/sgshim.acl

[A] To add file(s) to file list
[D] To delete files(s) from file list

[S] To backup files to target directory
[R] To restore files from target directory

[?] For Help
[Q] To quit the program
[X] Return to the Main Menu

Command ==>
```

1. **[T] Change target directory** The files listed will be saved to the target directory or device file which is defaulted to your floppy drive. You may change this to any directory or media you choose.
2. **[A] Add files to list/[D] Delete files from list** Use the add and delete options to adjust the file list accordingly.
3. **[S] Backup files** Press **S** to run the backup. Make certain the file list has been adjusted accordingly.

4. **[R] Restore files** When restoring files, all the files from the target directory will be restored to the SmartGate Server's root directory regardless of which files are listed.

## Uninstalling the Software

If you want to uninstall the SmartGate Server software press **U** in the Main Menu. The program will prompt you before completing the uninstall.

```
WARNING! If you proceed you will uninstall the
software!

Are you sure you want to uninstall the SmartGate
Server software? yes/ [no]

Do you want to remove SmartGate directories
(remove all directories with root /usr/smartgate)
yes/ [no] ?
```

**NOTE:** After the uninstall process is complete and if you removed all your SmartGate directories, you will need to `cd /` to a valid directory.

Under certain circumstances, you may encounter a file system corruption warning message. A system reboot will eliminate this message.

# Chapter 4

## Installing the SmartGate Server Software— Windows NT

**WARNING!** Your hard disk partition must be NTFS, not FAT.

**NOTE:** Your CD-ROM drive may differ depending on your system configuration.

**WARNING!** V-ONE highly recommends that you perform a complete backup before installing the new SmartGate Server software.

*Figure 4-1  
Service Detection Window*

This chapter provides instructions for installing the SmartGate Server software in a Windows NT environment.

### Installing the Software on Your System

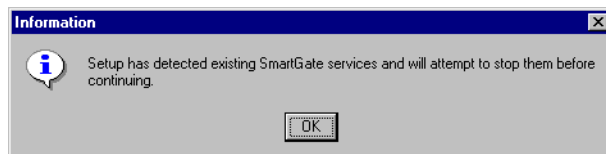
Follow the procedures in this subsection for both new installations and upgrades.

To install the SmartGate Server software:

1. Insert the SmartGate NT Server CD-ROM into your CD-ROM drive.
2. Click the Windows NT **Start** button and select **Run**.
3. Type **D:\SMTGate\NT\install\setup.exe** and click **OK**.

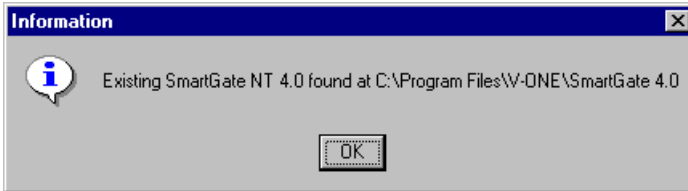
If you are upgrading your system from a previous version of SmartGate, you will see the following figures, otherwise proceed to step 4.

All running SmartGate services must be stopped before upgrading. Stopping the SmartGate services has been automated during the installation process (Figure 4-1). Click **OK** to continue.



A Welcome Window will be displayed, click **Next** to continue. A License Agreement Window is displayed next, accept the agreement and continue.

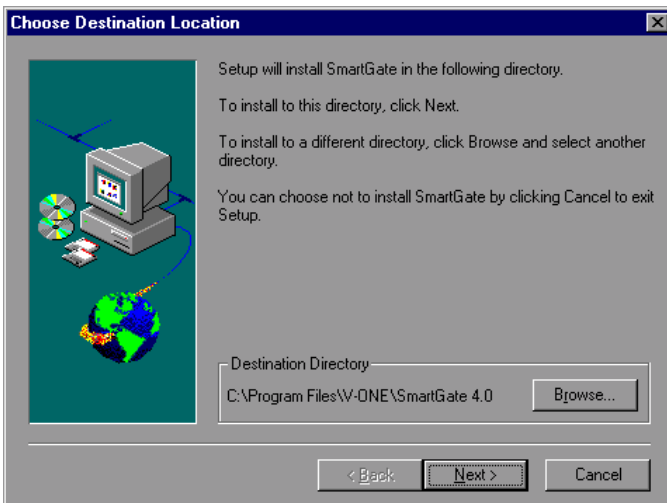
The installation process will determine if your computer system has a previous version of the SmartGate Server software installed. If it does, the installation program will inform you of its existence (Figure 4–2).



*Figure 4–2  
Upgrade Existing SmartGate  
Server Information*

Click **OK** to continue with the installation.

4. You will be asked to choose the destination of your SmartGate Server files before confirming the upgrade (Figure 4–3).

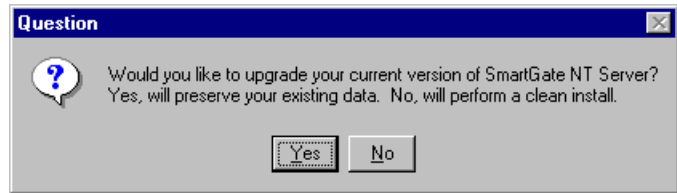


*Figure 4–3  
Choose Destination Location  
Window*

The Setup Wizard creates a default directory to store the SmartGate files, but you have the option to use the **Browse** button or manually enter any directory you wish. When finished, click **Next**.

5. If you are upgrading you will be asked to confirm the upgrade (Figure 4–4).

**Figure 4-4**  
**Upgrade Confirmation Window**

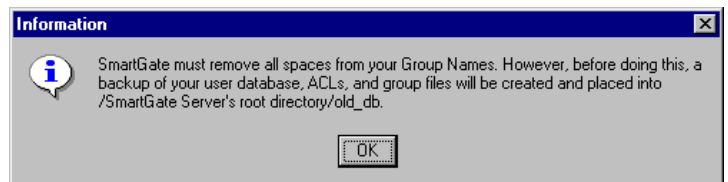


Click **Yes** to upgrade your existing version. The program will uninstall your existing SmartGate Server software. However, the configuration files, such as `reginfo.dat`, `sgconf.ini`, or any of the `.acl` or `.db` files, will be retained. They will be updated to reflect the new version, but the pertinent information will remain unchanged. During an upgrade installation, the new software is installed in the specified root directory.

However, if you click **No**, a new version of SmartGate will be installed in the specified directory, but without any of your previous configuration information, such as your `reginfo.dat`, `sgconf.ini`, or any of the `.acl` or your user database files.

6. If you are upgrading from a previous version of SmartGate, the program, `dbconv.exe`, will automatically search your user database, access control lists (ACLs), and group files and remove all spaces from SmartGate group names if necessary. Figure 4-5 will be displayed.

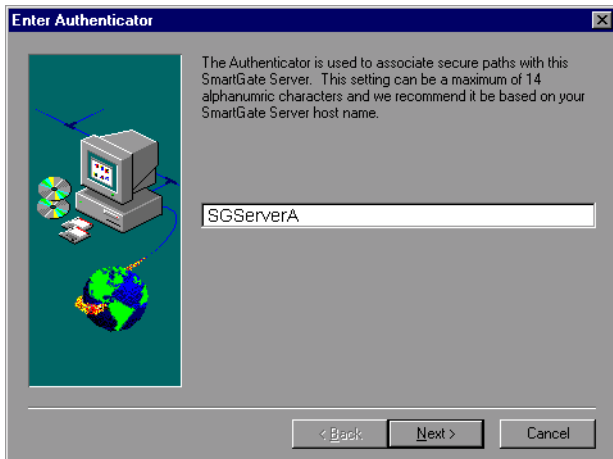
**Figure 4-5**  
**Remove Spaces From Group Names Window**



A backup will automatically be made of all the pertinent files and stored in `/SmartGate Server's root directory/old_db`.

If you are upgrading from SmartGate 4.x to SmartGate 4.x, the installation program will prompt you to reboot your system directly after uninstalling. This is due to the IPsec portion of SmartGate. You will be given the option of having the installation program automatically restart after reboot.

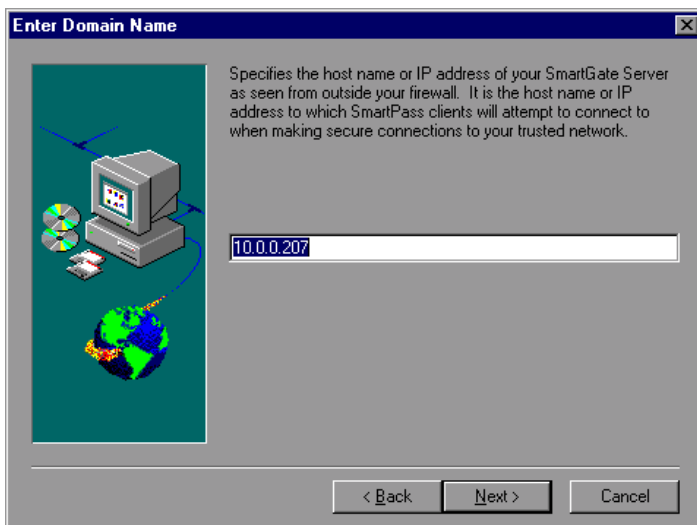
7. Figure 4–6 is displayed next.



**Figure 4–6**  
**Enter Authenticator Name**  
**Window**

**NOTE:** The Authenticator can be up to 14 alphanumeric characters in length.

Enter an **Authenticator** name (**authenticator**). The Authenticator is used to associate secure paths with the SmartGate Server. The installation program will insert the name of your machine as the default value; however, if you are upgrading, the Authenticator name as defined by your previous version will be inserted as the default value. Confirm the appropriate name and click **Next**. Figure 4–7 is displayed.



**Figure 4–7**  
**Enter Domain Name**  
**Window**

**WARNING!** For licensing reasons, please make certain that your host name is unique. Do not use a generic name like “SGServer.”

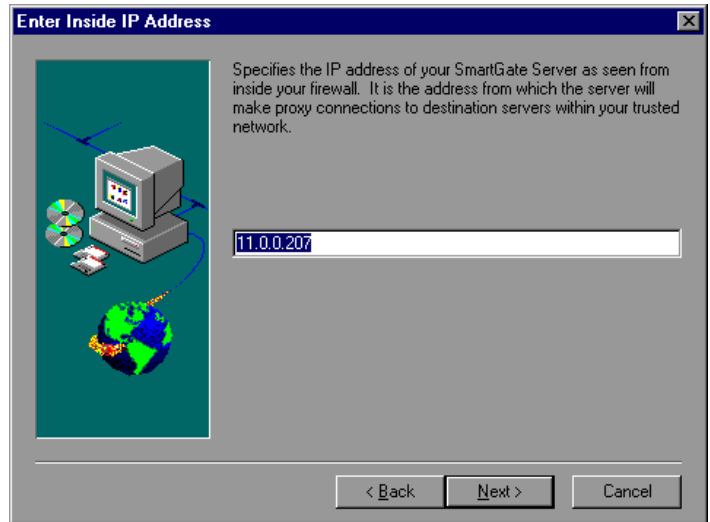
**WARNING!** To use IPSec features or UDP broadcasting (UDPPortList), an IP address—NOT a hostname—must be used for the Server name.

**WARNING!** For licensing, your hostname is case specific. For example, capital and lowercase letters can be used, but they must match exactly to the hostname in the license.



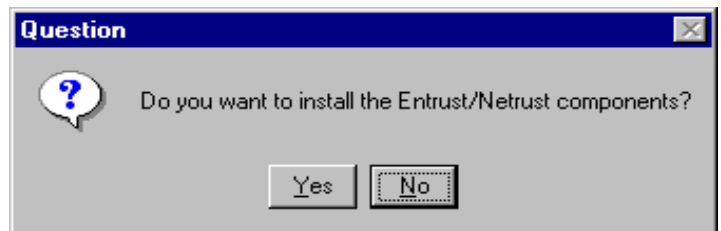
**Figure 4-8**  
**Enter Inside IP Address Window**

8. Enter either the hostname or IP address of the SmartGate Server (**domainname**). This allows SmartPass clients to locate the server from outside the firewall. If you are upgrading, the installation program will insert the current hostname as defined by your previous version. Click **Next**, Figure 4-8 is displayed.

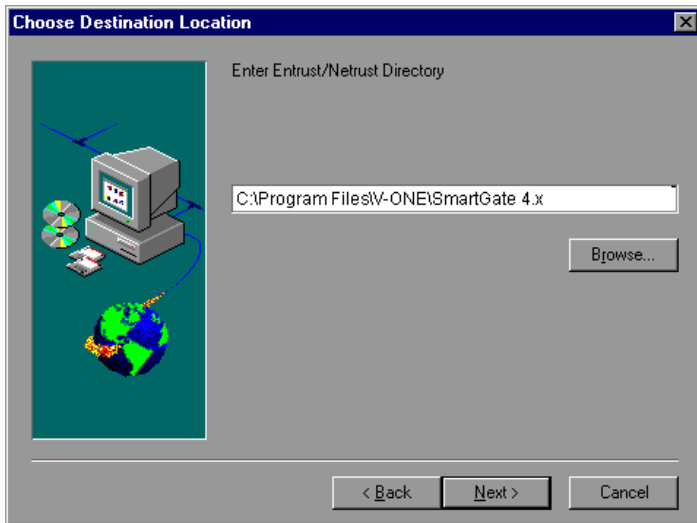


9. Enter your inside IP address (**InsideIP**). The SmartGate Server will use the inside IP address to make proxy connections to destination servers within your trusted network. If you are upgrading, the installation program will insert the current inside IP address as defined by your previous version. Click **Next** and Figure 4-9 is displayed.

**Figure 4-9**  
**Entrust/Netrust Installation Question**



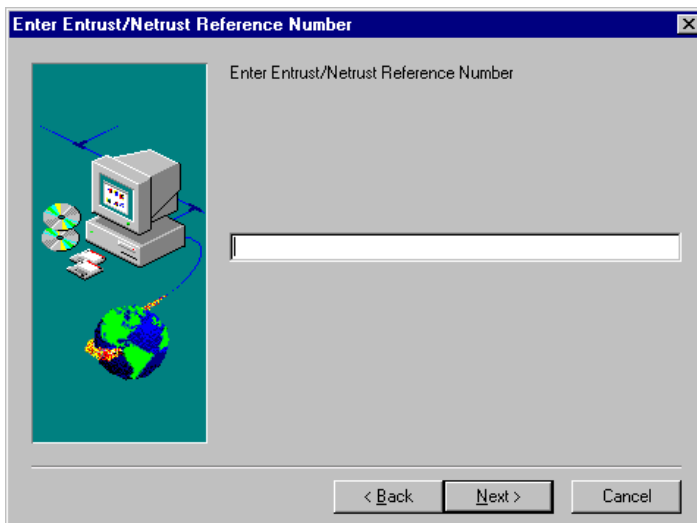
10. If you are using either Entrust or Netrust as SmartGate's authentication method and have already installed the required software, click **Yes**. Figure 4-10 will be displayed. If you are not using Entrust or Netrust authentication, click **No** and proceed to step 13.



**Figure 4-10**  
**Entrust/Netrust Authentication**  
**Enter Entrust Directory Window**

**NOTE:** The following three Entrust/Netrust Authentication windows are only applicable if you are using either Entrust or Netrust as SmartGate's authentication method.

11. Locate the directory where your Entrust/Netrust files (specifically `entrust.ini`) were installed. The `entrust.ini` file contains the location of the Entrust or Netrust CA Server and Manager and is used by both the SmartGate Server and SmartPass. Click **Next**, Figure 4-11 is displayed.



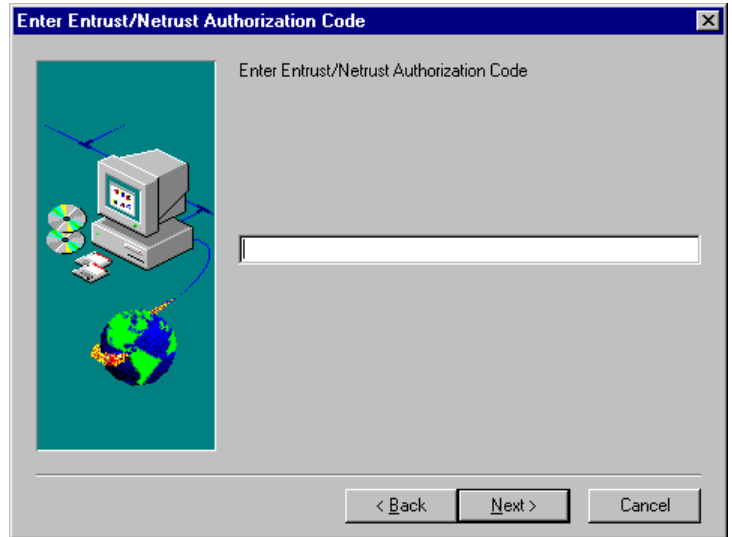
**Figure 4-11**  
**Entrust/Netrust Authentication**  
**Enter Entrust Reference Number Window**

**NOTE:** See your Entrust or Netrust documentation for complete information regarding your Entrust/Netrust Reference number and Authorization code.

12. Enter the Entrust/Netrust Reference number obtained from your Entrust or Netrust software. Click **Next**, Figure 4-12 is displayed.

**Figure 4-12**  
**Entrust/Netrust Authentication**  
**Enter Entrust Authorization Code**  
**Window**

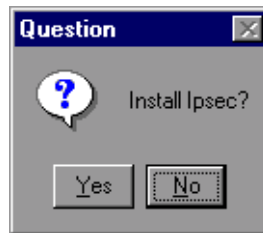
**NOTE:** Please refer to the *SmartGate With Netrust Authentication Guide* for detailed information on SmartGate/Netrust configuration options.



13. Enter the Entrust/Netrust Authorization code obtained from your Netrust software. Click **Next** and Figure 4-13 is displayed.

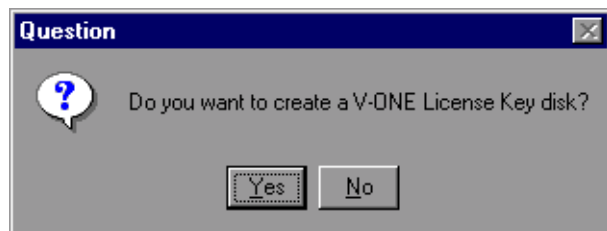
**Figure 4-13**  
**Install IPsec Question**

**NOTE:** For detailed instructions on configuring IPsec, see Chapter 11, "IPsec."

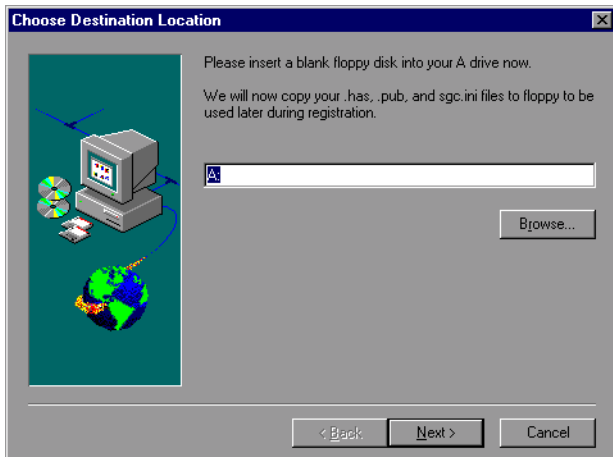


14. Click **Yes** if you want to install IPsec transport functionality on your SmartGate Server. Figure 4-14 is displayed.

**Figure 4-14**  
**Create V-ONE License Key**  
**Question**



15. Click **Yes** to verify that you do want to create a License Key disk. Figure 4-15 is displayed.



**Figure 4-15**  
**Choose Destination Location of Key Files Window**

**NOTE:** The only reason you would click **No** is if you are upgrading from SmartGate 4.x to SmartGate 4.x and already had a License Key and certificate in the appropriate directory.

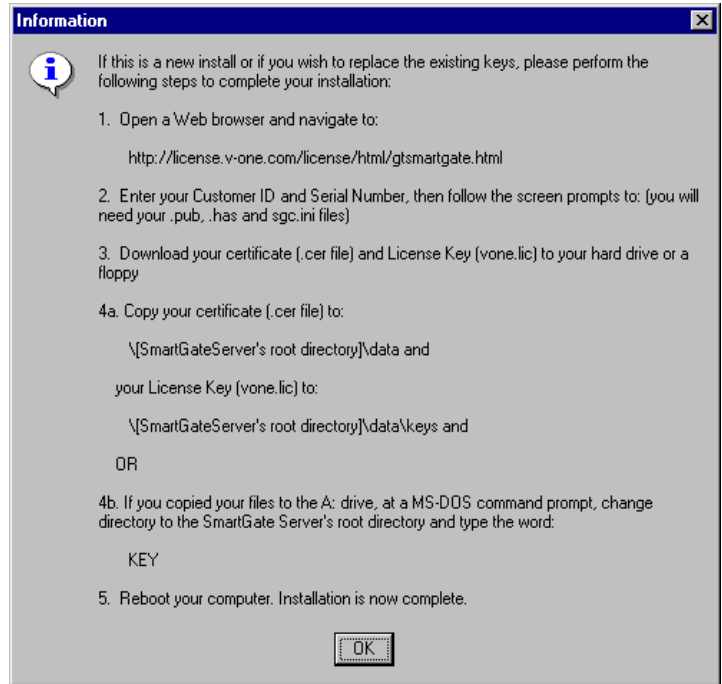
16. During the licensing process, you will be required to input the Soft Limit for the SmartGate License. The Soft Limit is the point at which the SmartGate administrator is notified that available user seats are running out. (For example, if you have a 100-seat license and have set a Soft Limit of 90, when the 90th user account is enabled, the administrator will be notified that there are 10 user seats available.)
17. The installation program will copy your .has, .pub, and sgc.ini file to the location of your choice. You will need these files when registering your SmartGate Server and obtaining your license and certificate. If you have access to the Web from the computer where your SmartGate Server resides, you may copy these files to your hard drive. However, if you need to move to another computer to access the Web you may want to copy them onto a floppy disk on your A:\ drive.
18. Figure 4-16 confirms that the files have been copied successfully and gives further instructions on obtaining your license.

Use the following section, “Obtaining Your License and Certificate” for step-by-step instructions of the process outlined in Figure 4-16.

**NOTE:** The sgc.ini file will always be created in the \SmartGate Server’s root directory\data.

**Figure 4-16**  
**Confirmation Window**

**WARNING!** SmartGate Server On-Line Registration will not work without receiving and installing the License Key and certificate from V-ONE.



## Obtaining Your License and Certificate

1. Register your SmartGate software if you have not already done so:
  - Open a Web browser and navigate to:  
<http://license.v-one.com/license/html/gtsmartgate.html>
  - Click **Register**, type in the requested information, and click **Submit**.
  - V-ONE will send a Customer ID and Serial Number to the e-mail address specified during registration.
2. Generate your License Key:
  - Open a Web browser and navigate to:  
<http://license.v-one.com/license/html/gtsmartgate.html>
  - Click **Generate License Key** and type in your Customer ID and Serial Number.

- Designate the license key Soft Limit notification. The Soft Limit is the point at which the SmartGate administrator is notified that available user seats are running out.
- Upload to V-ONE the *keyname.has*, *keyname.pub*, and *sgc.ini* files by using the browse buttons to select the location of each file.

If you created a License Key disk, copies of these files should be on the A:\ drive or to the location you specified in Figure 4–15. Otherwise, all of these files reside in C:\SmartGate Server's root directory\data.

### 3. Install your License Key:

After uploading the *keyname.has*, *keyname.pub*, and *sgc.ini* files, the Web page will display hyperlinks to download *vone.lic*, which is your License Key, and *keyname.cer*, which is your certificate, directly from the V-ONE Web site to the appropriate location.

- If your SmartGate Server has direct access to the Web:

Download the *keyname.cer* file to C:\SmartGate Server's root directory\data.

Download the *vone.lic* file to C:\SmartGate Server's root directory\data\keys.

- If your SmartGate Server does not have access to the Web, download both files to your License Key disk on your A:\ drive.

Open an MS-DOS Command Prompt and change your directory to the SmartGate Server's root directory. For example:

**cd C:\Program Files\V-ONE\SmartGate 4.x\**

Type: **KEY** and press ENTER.

Your certificate and License Key are copied from your A:\ drive and installed into their appropriate location.

You cannot generate a License Key or a certificate more than once. If you want to buy additional seats, you must contact V-ONE. We will reset the total number of seats with the additional seats added. You may then revisit our Web site and, using the same Customer ID and Serial Number, generate a new License Key.

Installation of the SmartGate Server software is now complete. You must reboot your system before proceeding.

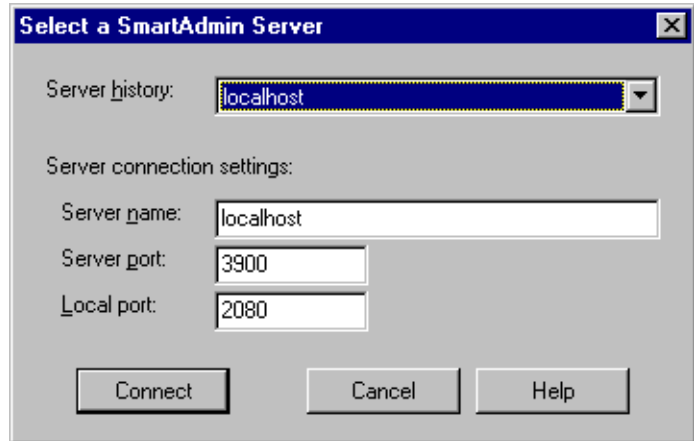
**NOTE:** You must use the browse button, you cannot type in the location of the file. If your Web browser software does not support the upload functionality and the browse buttons do not appear, you will need to upgrade your browser.

**NOTE:** To display your complete license information at any time on a Microsoft Windows NT Server, run the *vladmin.exe* file in C:\SmartGate Server's root directory from an MS\DOS Command Prompt type **vladmin**.

*Figure 4-17  
Select a SmartAdmin  
Server Window*

## Launching SmartAdmin

To open SmartAdmin, SmartGate's administrative GUI, click **Start** and **Programs**; then select **SmartAdmin**. Figure 4-17 is displayed.



Since you are running SmartAdmin locally (at your server console), enter `localhost` for the Server Name, **3900** for the Server Port, and then connect.

Use SmartAdmin to configure your SmartGate Server, i.e., create Web and TCP Access Permissions, enter On-Line Registration settings, etc. See [Chapter 5, “Using SmartAdmin”](#) for complete information.

## Adding and Removing Services

A SmartGate service defines one encrypted proxy type (i.e., SmartGate-encrypted FTP). SmartGate services are automatically added and removed at the appropriate times. However, if for some reason you need to add or remove a SmartGate service, use the following instructions.

1. Stop the SmartGate service (Go to **Control Panel, Services, SmartGate Server**, and click **Stop**).
2. Open a command prompt and change to the SmartGate Server directory.
3. For the services listed in Table 4-1 that you want to add, enter the command:

***prog\_name -regserver service***

where *prog\_name* is the program name to be entered.  
*service* is the service you wish to start.

<i>prog_name</i>	<i>service</i>	Description
<b>servers</b>	sgate	Generic TCP proxy
<b>servers</b>	sweb	Web proxy
<b>servers</b>	sgftp	FTP proxy (multi-channel)
<b>servers</b>	spsgftp	FTP proxy (single-channel)
<b>servers</b>	sgasrv	Authentication Server
<b>servers</b>	sgreg	On-Line Registration Server
<b>servers</b>	sgora	SQLNet/Oracle proxy
<b>servers</b>	sguidsrv	User ID Server
<b>servers</b>	sgccag	Dynamic Configuration agent
<b>servers</b>	sgccsrv	Dynamic Configuration Server
<b>servers</b>	adm_gw	Remote administration
<b>servers</b>	sgadmin	SmartAdmin Server
<b>servers</b>	sgrdb	Redundant Database Server

For example:

**servers -regserver sweb**

4. For the services listed in Table 4–2 that you want to add, enter the command:

***prog\_name* -regserver**

where *prog\_name* is the program name to be entered.

<i>prog_name</i>	Description
<b>sgproxy</b>	Single Port Proxy
<b>sgsdi</b>	SmartGate/SecurID Server
<b>sgradius</b>	SmartGate/RADIUS Server
<b>sglic</b>	License manager
<b>sgent</b>	SmartGate/Entrust/Netrust Server

For example:

**sgradius -regserver**

5. Start SmartGate (Go to **Control Panel, Services, SmartGate Server**, and click **Start**).
6. To remove services, use the command **-unregserver** instead of **-regserver**.

**Table 4–1**  
**Program Services and Names**

**Table 4–2**  
**Additional Program Services and Names**



# Chapter 5

## Using SmartAdmin

SmartAdmin provides a convenient way to manage user information and access permissions for remote hosts and World Wide Web services. With SmartAdmin you can:

- Create and maintain users and groups
- Enable/disable users
- Create and maintain both TCP and Web access permissions
- Create and maintain IPSec access permissions, channels, and site-to-site configuration
- Configure the [On-Line Registration \(OLR\)](#) settings to capture data appropriate to your company's identification and processing requirements
- Assign administrative privileges
- Set SmartGate Server configuration parameters including: user authentication variables and [Single Port Proxy](#) rules
- Administer PKI CA Server Certificates and attributes

There are four levels of SmartGate administrative privileges. The administrative level controls an administrator's privileges regarding users and groups.

**Minimal** Administrators at this level can only enable/disable users and edit a user's name in the event of a name change or a typographical error. Administrators at this level may also be restricted to administering only certain groups.

**Restricted** In addition to those rights provided at the minimal level, administrators can change authentication keys and add/edit/delete end users. Access at this level may be limited to certain groups.

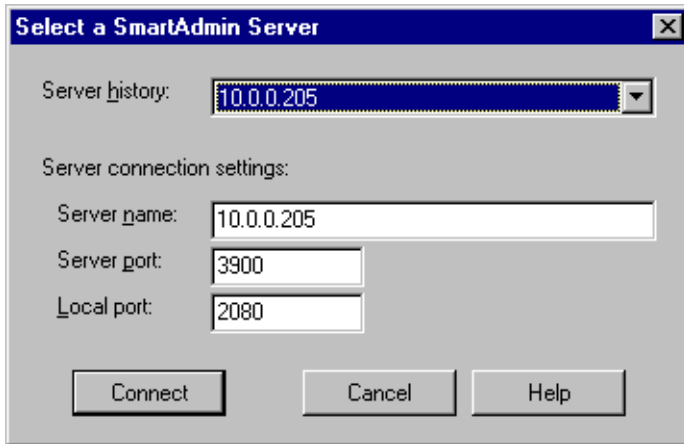
**Standard** In addition to those rights provided at the restricted level, administrators can add/edit/delete all access permissions. Access at this level may be limited to certain groups.

**Superuser** Administrators at this level have access to all settings. In addition to the privileges of the standard administrator, superusers can assign administrator levels, change

SmartGate configuration settings, and configure the OLR, IPSec Channels, PKI, and Single Port Proxy Map files. Superusers have full access to all groups.

## Launching SmartAdmin

After launching SmartAdmin, the system will display the window shown in Figure 5–1.



*Figure 5–1  
Select a SmartAdmin Server  
Window*

Use this window to designate your SmartGate Server. Administration can be performed either locally on a Windows NT SmartGate Server or remotely on a personal computer.

### ■ Locally:

When running SmartAdmin locally (at your server console) enter `localhost` for the Server name and **3900** for the Server port.

### ■ Remotely:

By default, the SmartGate Administration Server listens to port 3900 and the SmartGate Web server to local port 2080. To run SmartAdmin on a remote computer secured by SmartGate, fill in your fully qualified **Server name**, and enter **3900** in the Server port field and **2080** in the Local port field. SmartPass must be running before SmartAdmin can be launched remotely.

All connections to SmartGate Servers are placed in the **Server history** drop-down list. When you select a server from the drop-down list, SmartAdmin automatically fills in the values for

**NOTE:** If your SmartGate Server is running Windows NT, SmartAdmin may be run locally with superuser administrative privileges. However, to use SmartAdmin remotely, you still must register and add yourself to the `adm-gw.acl` file under the Admin Rights tab.

that server. If your server is not in the drop-down list, type the appropriate information in the fields. When the values are correct, click **Connect**.

## Preliminaries for Remote Administration

Before you can launch SmartAdmin and perform administrative functions, you must:

1. Install the SmartPass software on your personal computer.
2. Check that the smart card is in the reader (if a smart card is being used).
3. Perform OLR.
4. Enable your User ID and give yourself administrative privileges.

## Setting Yourself Up as an Administrator

Before you can run SmartAdmin remotely, you must:

1. Add yourself to the user database by performing OLR.
2. Enable your User ID.

### UNIX-based:

From the console, change your directory to the SmartGate Server's root directory and type:

**`./sgadm -enable user`**

where: *user* is your User ID.

### Microsoft Windows NT:

- Start SmartAdmin and click the **Users** tab.
- Select your user record, click **Edit**, and select the **Enabled** check box.

3. Give yourself administrative privileges by adding your User ID, access level, and accessible groups to the `adm-gw.acl` file.

The format of the administrative privileges line in `adm-gw.acl` is:

**`user level group1,group2,...`**

where: *user* is your User ID.

*level* is the type of SmartGate administrator:  
minimal, restricted, standard, or superuser.

*group(s)* is the group(s), delineated by commas, to which the administrator will have access or **any**, which includes all groups.

**NOTE:** The `adm-gw.acl` file maximum line length is 256 characters.

**NOTE:** The `adm-gw.acl` file is located in the SmartGate Server's root directory/etc on a UNIX-based server and SmartGate Server's root directory\data on Windows NT.

There are multiple ways to edit `adm-gw.acl`:

#### **UNIX-based:**

- Use the installation script from the SmartGate Server software by running `./setup` from /SmartGate Server's root directory/bin.
- Edit the file using a text editor.

#### **Microsoft Windows NT:**

- Start SmartAdmin and click the **Admin Rights** tab.
- Edit the file using a text editor.

At this point, you, as the [SmartGate Server administrator](#), have superuser administrative privileges. You can now launch SmartAdmin and begin administering your SmartGate System.

## **Basic Usage**

The SmartAdmin window contains a menu bar and a status bar, both of which are visible from every page. The menu bar contains the following menu options for editing and manipulating data on the current page.

**File** - Exit SmartAdmin or specify a new SmartGate Server

**Edit** - Select, add, edit, or delete table entries; select all and jump to selected entries

**View** - Access the Find, Find Next, and Filter commands to perform searches; refresh and reload table entries

**Help** - Access online help

Along the bottom of the SmartAdmin window is the status bar. The status bar is divided into two sections or "panes." The left pane displays informational messages while SmartAdmin is performing an action. The right pane displays the number of entries in the current table and how many of these entries are currently selected.

SmartAdmin uses many of the features common to Windows programs. For example:

- Select multiple entries by holding down either the **Shift** or **Ctrl** key and clicking on the desired records.
- Click a record with the right mouse button to open a popup menu containing most of the options available from the menu bar.
- Open an Edit Window by double-clicking on a record.

**Figure 5-2**  
**Users Table**

**NOTE:** The IPSEC Access and IPSEC Channels tabs are explained in detail in Chapter 11, “IPSec.”

**NOTE:** If you have an established user database, those User IDs can be used for your SmartGate user database. This field can be up to 30 characters.

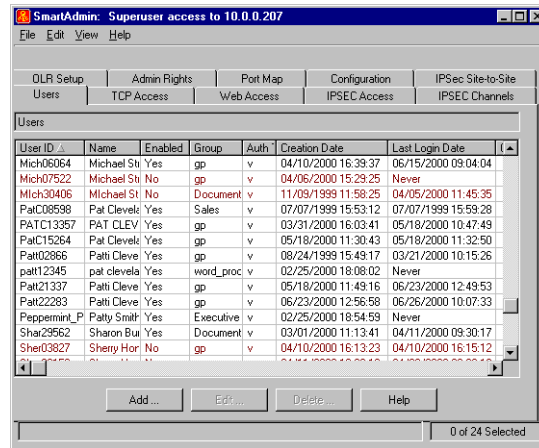
**NOTE:** User long names are case-sensitive.

**NOTE:** If no other group was assigned to a user, **gp** will be listed as their group.

**NOTE:** Group names can be up to 23 characters, are case-sensitive, and cannot contain spaces or special characters (\*,/,).

## Managing Users

The Users Table (Figure 5-2) displays the contents of the SmartGate user database.



User ID	Name	Enabled	Group	Auth	Creation Date	Last Login Date
Mich06064	Michael Stu	Yes	gp	v	04/10/2000 16:39:37	06/15/2000 09:04:04
Mich07522	Michael Stu	No	gp	v	04/06/2000 15:29:25	Never
Mich30406	Michael St	No	Document	v	11/09/1999 11:58:25	04/05/2000 11:45:35
PatC08598	Pat Clevele	Yes	Sales	v	07/07/1999 15:53:12	07/07/1999 15:59:28
PATC13357	PAT CLEV	Yes	gp	v	03/31/2000 16:03:41	05/18/2000 10:47:49
PatC15264	Pat Clevele	Yes	gp	v	06/18/2000 11:30:43	05/18/2000 11:32:50
PatC02965	Patti Clevele	Yes	gp	v	08/24/1999 15:49:17	03/21/2000 10:15:26
patt12345	pat clevele	Yes	word_proc	v	02/25/2000 18:08:02	Never
Pat21337	Patti Cleve	Yes	gp	v	05/18/2000 11:49:16	06/23/2000 12:49:53
Pat22283	Patti Cleve	Yes	gp	v	06/23/2000 12:56:58	06/26/2000 10:07:33
Peppermint_P	Patty Smith	Yes	Executive	v	02/25/2000 18:54:59	Never
Sha29562	Sharon Bui	Yes	Document	v	03/01/2000 11:13:41	04/11/2000 09:30:17
Shed03827	Sherry Hor	No	gp	v	04/10/2000 16:13:23	04/10/2000 16:15:12

Each user record contains the following fields:

- User ID** Unique identifier for every user.
- Name** Typically contains the user's first and last names separated by a space, also referred to as the user's long name. (If using OLR, this field is automatically created from the first two fields.)
- Enabled** To allow authentication of new users. Users can be automatically enabled after completion of OLR, or they can be manually enabled by an administrator at a later date. Records for disabled users are shown in red.
- Group** Users are given all of the access permissions that apply to their group.
- Auth Type** The authentication type utilized.
- Creation Date** Date and time of registration.
- Last Login Date** Date and time that the user most recently accessed the SmartGate System.
- Other Info** Displays the pager information.

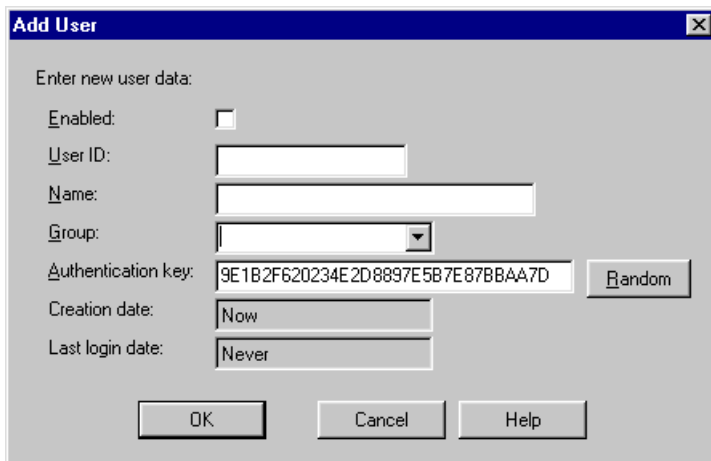
The Users Table provides the means to control the following functions:

- Add user
- Edit user
- Delete user
- Find users
- Filter users

The Add, Edit, and Delete commands are available from either the Edit menu or the buttons at the bottom of the Users Table. To restrict the data that is downloaded and displayed, click **View** on the menu bar, and then click **Filter**. You can also open a shortcut menu containing these commands by selecting a user record and clicking the right mouse button.

## Add User

Most user records are created during OLR. If you are not using OLR or a predefined user database and you need to add a user, click **Add** on the Users Table and Figure 5-3 is displayed.

The image shows a Windows-style dialog box titled "Add User". It contains several input fields and buttons. The fields are: "Enabled:" with a checkbox, "User ID:" with a text box, "Name:" with a text box, "Group:" with a dropdown menu, "Authentication key:" with a text box containing the value "9E1B2F620234E2D8897E5B7E87BBAA7D" and a "Random" button next to it, "Creation date:" with a text box containing "Now", and "Last login date:" with a text box containing "Never". At the bottom are three buttons: "OK", "Cancel", and "Help".

**Figure 5-3**  
**Add User Window**

**WARNING!** If you add a user using the Add User function, you must have the end user format his/her token in order to store the authentication key on it. See “Adding/Changing your Authentication Key” in Chapter 5, “Installing and Registering SmartPass for Microsoft Windows,” of the *SmartPass Administrator’s Guide*.

Enter the following information to define the new user:

1. **Enabled** Select this check box to allow the user to access SmartPass immediately after registration.
2. **User ID** A unique identifier created for the user.
3. **Name** The user’s full name.
4. **Group** The group to which the user is assigned (other than “all”).

**WARNING!** If you change the end user's authentication key, the user will need to store the new authentication key on their token. See "Adding/Changing your Authentication Key" in Chapter 5, "Installing and Registering SmartPass for Microsoft Windows," of the *SmartPass Administrator's Guide*.

**Figure 5-4**  
**Edit User Window**

**NOTE:** For security reasons, the current authentication key will not be displayed.

5. **Authentication Key** The user's secret authentication key is a 32-bit hexadecimal key assigned to each SmartGate user. This key, which is stored on the user's smart card or virtual token and in the SmartGate Server's user database, consists of any arrangement of digits 0 through 9 and uppercase letters A through F. The value of the key is not important, as long as it is kept secret. On the **Add User** window, a randomly generated key is displayed as a default, but you can enter any key sequence or generate a new one by clicking **Random**.
6. **Creation Date/Last Login Date** You cannot edit these fields. Information in these fields is supplied automatically.

## Edit User

The Edit User command allows administrators to:

- Correct a user's name or group
- Change a user's secret authentication key
- **Enable/disable** a registered user
- View a user's OLR data

Select the desired user record and click **Edit** or double-click the user record. Figure 5-4 will be displayed.

The screenshot shows the 'Edit User' dialog box. It contains the following fields and sections:

- Selected user:**
  - User ID: Patt02866
  - Creation date: 08/24/1999 15:49:17
  - Last login date: 08/25/1999 12:06:35
- On-Line Registration data:**

Name	Value
First Name	Patti
Last Name	Cleveland
E-mail	pcleveland@v-one.com
- Edit user:**
  - Enabled: ☒
  - Name: Patti Cleveland
  - Group: Finance (dropdown menu)
  - Authentication key: [Redacted] Random
- Buttons: OK, Cancel, Help

To edit user information, select the field and enter the appropriate data.

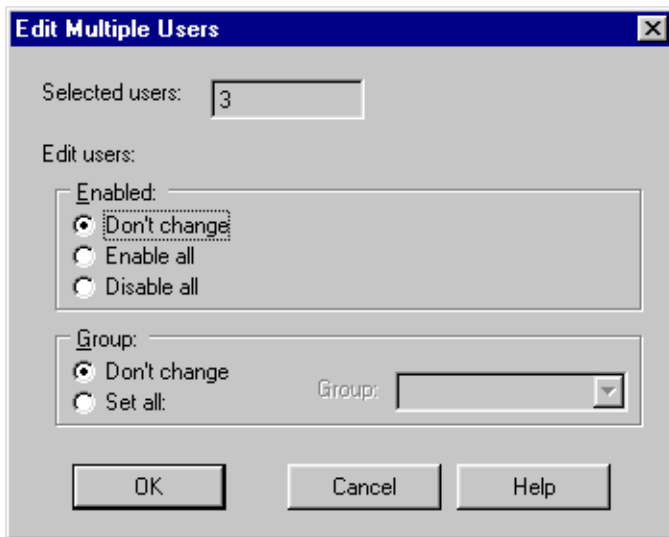
If the user was created through OLR, the **Edit User** window will contain her OLR data. However, this section may be blank if the users are created using some other method, such as SmartAdmin. Since the OLR data structure can be modified, users may also have a variety of different information.

## Delete User

If you highlight a user record and click **Delete**, SmartAdmin will list all information about the selected user and confirm that you want to delete that user.

## Multiple User Management

When editing and deleting users, SmartAdmin allows you to work with multiple user records. To edit a number of user records simultaneously, select the desired user records on the Users Table and click **Edit**. Figure 5-5 is displayed.



**Figure 5-5**  
**Edit Multiple Users Window**

**NOTE:** From the Edit Multiple Users Window you can only select the Enabled and Group commands.

When deleting multiple users, select the desired records and click **Delete**. The system will confirm deletion of each user separately.

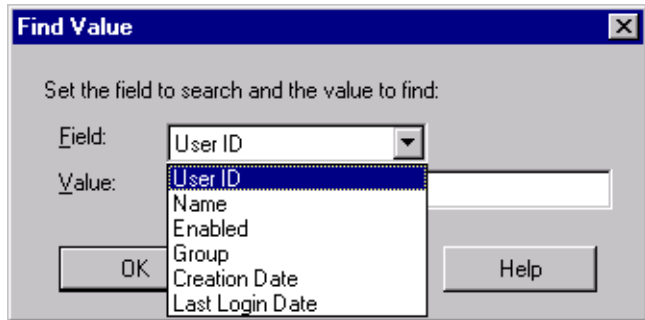
**WARNING!** You may prefer to disable a user rather than delete her. If you disable a user, the user's record turns red and the Enabled column is marked **No**. User information is retained, but the disabled user is not counted as an active user in your licensing agreement.



## Find and Find Next

To search for a specific user record or access permission, click **View** on the SmartAdmin menu bar, point to **Find** or **Find Next**, and release the pointer. Figure 5-6 is displayed.

*Figure 5-6  
Find Value Window*

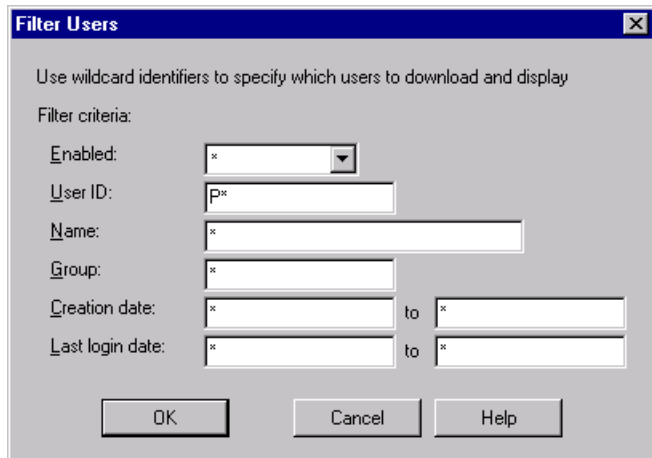


Select from the drop-down list which field you want to search, and then type the words you want to search for in the **Value** text box. Wildcard strings can be used as search parameters. For example, by selecting **User ID** and then entering 'B\*' in the **Value** text box, you are requesting a search for only those users whose User ID begins with the letter 'B'.

## Filter Users

To display a specific set of user records, click **View** on the SmartAdmin menu bar, point to **Filter Users**, and release the pointer. Figure 5-7 is displayed.

*Figure 5-7  
Filter Users Window*



**NOTE:** Your filtered display will be retained on restarting SmartAdmin.

**NOTE:** Use **Refresh** in the **View** menu to update the filtered display and **Reload All** to reload the entire user database.

The Filter Users Window allows you to set restrictions on which user records are downloaded and displayed. Wildcard strings

can be used to set the restrictions. For example, by entering 'B\*' in the User ID text box, you are requesting a display of only those users whose User ID begins with the letter 'B.' The date text boxes (**Creation Date** and **Last Login Date**) are a special case. You can specify either a range of dates or the wildcard '\*' to indicate no limit. SmartAdmin will not recognize a wildcard string within a date, such as '10/\*/97.'

## Common User Management Tasks

A primary use of SmartAdmin is to enable and disable users, once they have correctly identified themselves and registered. SmartAdmin has several features that simplify this task.

1. To find all recently registered users; use the **Filter Users** command, set **Enabled** to **No**, and then type in a recent **Creation Date**.
2. To enable multiple users quickly; select those users you want to enable, click **Edit**, and select **Enable All**.
3. To find information about a specific user, such as his or her last login time or OLR data, given only the User ID or name; sort the user records on the User ID, Group, or Name by clicking on the column title, and then scroll down until you find the right record. Alternately, you can use the **Filter** command to display only that user's record.

## TCP and Web Access Permissions

SmartAdmin allows you to manage access permissions for users and groups at both the TCP and World Wide Web levels. Although SmartAdmin contains a separate table for each of these levels, the functionality of the two tables is virtually identical and are both described in this section.

Access permissions are defined as the associations between users and connections. As the SmartGate Server administrator, you are responsible for establishing and maintaining access permissions for your user community. By default, SmartGate implements the security policy: **"Access is forbidden unless explicitly permitted."** Using SmartAdmin, you will either grant or deny access to an individual user or to a group of users according to their ability to be authenticated.

**NOTE:** The IPSEC Access and IPSEC Channels tabs are explained in detail in [Chapter 11, "IPSec."](#)

**WARNING!** To create a valid new group, you must use SmartAdmin or command line; do not edit a file directly.

The permissions which control access are identified by their hostname or IP address, and by their port. SmartGate categorizes access permissions into two groups:

- TCP permissions (FTP, Telnet, etc.)
- Web permissions, which consists of a Web server (host) and a port

In both cases, permission is given to either a user or a group to access a single destination.

Every SmartPass user is assigned to one group, in addition to the group “all.” They receive their permissions through Dynamic Configuration each time they start SmartPass, the client software in the SmartGate System, and at regular intervals as defined by the user. Users receive these permissions from the following sources:

1. Explicitly assigned to the user
2. Assigned to the user’s group
3. Assigned to any included subgroups
4. Assigned to the group “all”

Assigning permissions to groups avoids unnecessary duplication and is more efficient than assigning permissions to individual users.

## Defining Access Permissions

SmartGate access permissions are defined as user permissions or group permissions. Group permissions are typically designed to assemble permissions in ways that are most useful to an organization. Aggregating permissions geographically, for example, works well for an organization that is geographically dispersed. The Chicago and New York LANs are examples of permissions grouped geographically.

The use of groups makes it possible to bundle permissions together. For example, the URLs to three Web pages could be assigned to the Web group ~Salesinfo. ~Salesinfo could then be assigned to the group ~Sales, which contains a number of permissions of its own. A user assigned to ~Sales would then have access not only to the permissions in that group but to the permissions in ~Salesinfo as well. To extend the scenario, if ~Sales were assigned to ~Marketing; then a user assigned to ~Marketing would have access to the permissions in his own group, in ~Sales, and in ~Salesinfo.

Example:

Let’s suppose that you have established the following groups for your company and you have an employee, **Joe**, in the **Marketing** Department (Figure 5–8).

[joe]	[~Salesinfo]	[~Marketing]	[~Sales]	[~all]
/xxx.xxx.com/	/xxx.xxx.com/	/xxx.xxx.com/	/xxx.xxx.com/	sql.v-one.com 1521
15.0.0.290	/xxx.xxx.com/	/xxx.xxx.com/	/xxx.xxx.com/	ftp.v-one.com 21
	/xxx.xxx.com/	~sales	~salesinfo	telnet.v-one.com *
				/*.v-one.com/

**Figure 5–8**  
**Access Permission Structure**

In the above example, Joe receives the permissions assigned directly to him as an individual user, all of the permissions in the ~Marketing group (because he is a member of that group) and all of the permissions in the group ~all. Note that as a member of ~Marketing, Joe receives the access permissions under ~Sales as well. He also receives the group permissions under ~Sales (e.g., ~Salesinfo).

Use caution when assigning multiple nesting groups. Remember to keep it simple. Start by planning your access permission environment based on your company’s security policy. Wildcards in TCP and Web access permissions should be used sparingly. For more information on wildcarding see [Appendix C, “ACL Wildcarding.”](#)

See [Chapter 6, “User Authentication”](#) for more information on how the Authentication Server reads the access permission files, `sgate.acl` and `sweb.acl`.

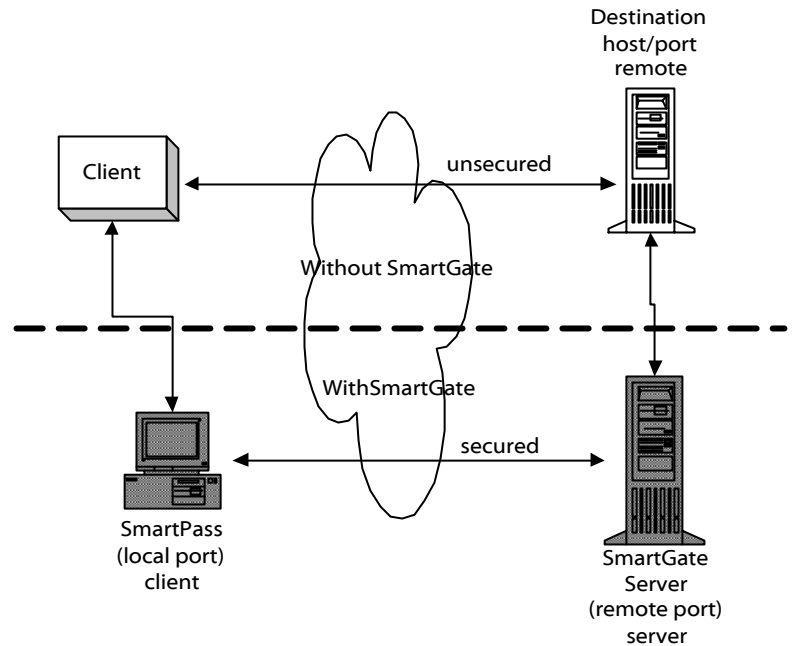
### TCP Access Permissions

Telnet, FTP, SMTP, and NNTP are all TCP-based protocols.

The destination hostname and port number specify a standard, unsecured connection to a remote host. SmartGate uses a server proxy (`sgate`) and a client proxy (SmartPass) to secure these communications. These two additional port numbers are required to specify the client and server proxy connections (Figure 5–9).

**Figure 5-9**  
**TCP Access**

**NOTE:** Requests from SmartPass to the SmartGate Server are sent through the Single Port Proxy (defaulted at 3845) and then routed to the appropriate remote port. The Single Port Proxy is available in SmartGate version 2.5 and later.



Several common TCP protocols, such as Telnet and FTP, have predefined default combinations of remote (destination), client, and server ports. Table 5-1 illustrates some of the most common defaults.

**Table 5-1**  
**TCP Ports**

Service	Destination Port	Server Port (rport)	Client Port (lport)
Telnet	23	2023	2023
FTP	21	2021	21
SMTP	25	2023	25
POP3	110	2023	110
NNTP	119	2023	119
Oracle	1521	3521	1521

As the SmartGate administrator, you will use the TCP Access Table (Figure 5-10) to furnish the access data for all secure sessions.

SmartAdmin: Superuser access to 10.0.0.207

File Edit View Help

Users TCP Access Web Access IPSEC Access IPSEC Channels OLR Setup Admin Rights Port Map Configuration

TCP Access Permissions

Owner Type	Owner ID	Perm. Type	Destination Host / Group	Service	Destination Port	Server Port	Client Port
Group	krakit	Path	127.0.0.1	Other	3848	2023	3848
Group	Sales	Path	92.0.0.100	Telnet	23	2023	2023
Group	Sales	Path	ABCServer.fence.company	FTP	21	2021	21
Group	all	Path	12.0.0.111	POP3	110	2023	110
Group	Management	Include	Sales				
Group	Management	Include	Finance				
Group	Management	Include	krakit				
Group	Finance	Path	12.0.0.*	Telnet	23	2023	2023

Add ... Edit ... Delete ... Help

0 of 8 Selected

Figure 5-10  
TCP Access Table

**NOTE:** The TCP Access permissions table reflects information found in the `sgate.acl` file located in the SmartGate Server's root directory on a UNIX-based server and SmartGate Server's root directory\data on Windows NT.

Each TCP access record (row) in the TCP Access Table displays a single permission assigned to either a group or an individual user. Each record contains either four or eight fields, depending on whether an access permission is a primary path or is included as subgroup/path. The fields requiring information are:

1. **Owner Type**

Specifies whether this permission is assigned to a user or a group.
2. **Owner ID**

Either the User ID, the name of the group that will receive this permission, or "all." All users belong to the universal group "all."
3. **Permission Type**

The value for this field can be either *Path* or *Include*. *Path* indicates that a path to a remote host and access permission has been designated. *Include* indicates that the specified Owner (either User or Group) will acquire the paths of the included group. Included group permissions display in green, while individual path permissions display in black.

**NOTE:** Wildcarded IP addresses, DNS names, and destination ports are valid. For more information, see Appendix C, “ACL Wildcarding.”

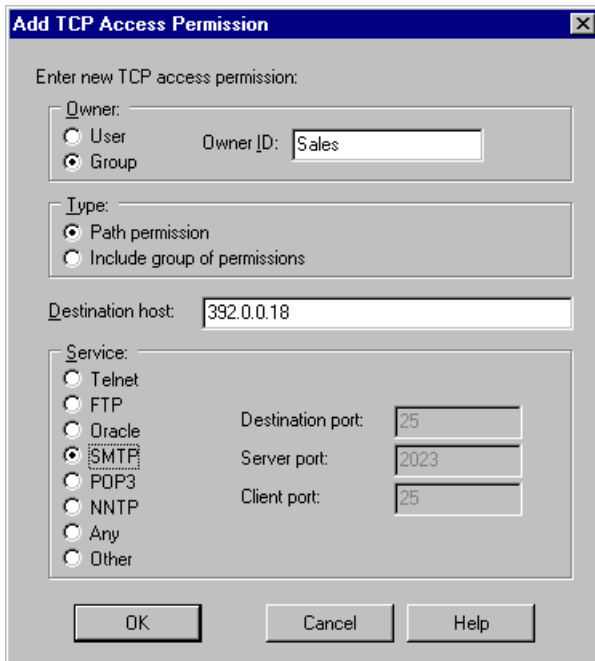
- |                                  |   |
|----------------------------------|---|
| 4. <b>Destination Host/Group</b> | Either the hostname or IP address of the path's destination host, or the name of the included group.  |
| 5. <b>Service</b>                | SmartGate has several predefined combinations of destination, server, and client ports (e.g., FTP, Telnet, Oracle, etc.). This field indicates either a selected combination or “Other.” Only when <b>Other</b> is selected is it necessary to specify the ports. |
| 6. <b>Destination Port</b>       | Destination port of the path (i.e., remote port).   |
| 7. <b>Server Port</b>            | SmartGate Server proxy port. The list of server ports can be found in the Port Map tab.   |
| 8. <b>Client Port</b>            | Local proxy port.   |

When managing TCP access permissions, four commands are available:

- Add an access permission
- Edit an access permission
- Delete access permissions
- Filter display of access permission records

### **Add/Edit TCP Access Permissions**

To add or edit TCP access permissions, use the command buttons at the bottom of the window or the menu bar options. Figure 5–11 is an example of the Add TCP Access Permission Window. The Edit TCP Access Permission Window contains the same information in a nearly identical layout.



**Add TCP Access Permission**

Enter new TCP access permission:

Owner:  
☐ User  
☒ Group  
 Owner ID: Sales

Type:  
☒ Path permission  
☐ Include group of permissions

Destination host: 392.0.0.18

Service:  
☐ Telnet  
☐ FTP  
☐ Oracle  
☒ SMTP  
☐ POP3  
☐ NNTP  
☐ Any  
☐ Other

Destination port: 25  
 Server port: 2023  
 Client port: 25

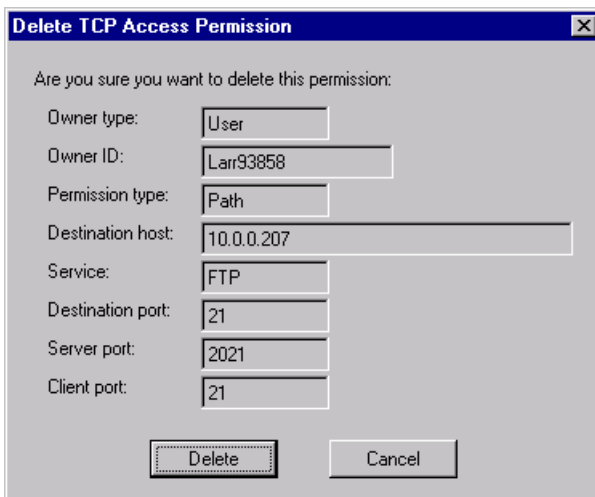
OK Cancel Help

**Figure 5-11**  
**Add TCP Access Permission**  
**Window**

The Add/Edit TCP Access Permissions Windows are self-explanatory—using the field descriptions on the preceding page—fill in the blanks with the appropriate information.

## Delete TCP Access Permissions

To delete TCP access permissions, select an access permission record or multiple records and either click **Delete** at the bottom of the window or use the menu bar option. Figure 5-12 is displayed.



**Delete TCP Access Permission**

Are you sure you want to delete this permission:

Owner type: User  
 Owner ID: Larr93858  
 Permission type: Path  
 Destination host: 10.0.0.207  
 Service: FTP  
 Destination port: 21  
 Server port: 2021  
 Client port: 21

Delete Cancel

**Figure 5-12**  
**Delete TCP Access Permission**  
**Window**

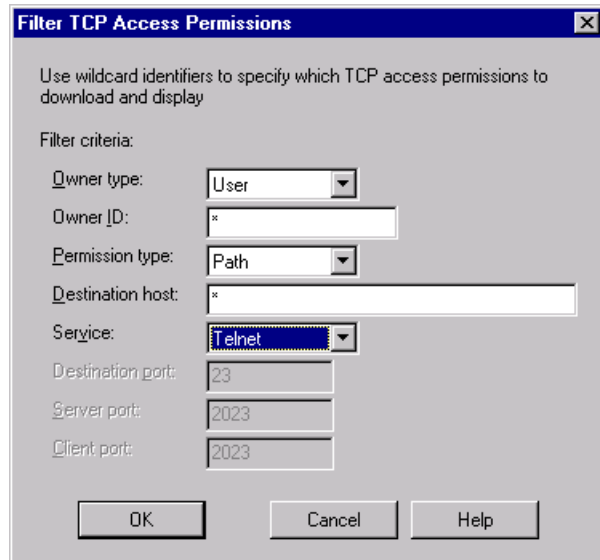


The Delete TCP Access Permission Window is a confirmation that you want to delete the permission. If you select multiple permissions for deletion, the system will ask for a deletion confirmation on each permission.

## Filtering

To display specific TCP access permission records, select **View** in the menu bar, and then select **Filter TCP Access Permissions**. Figure 5–13 is displayed.

*Figure 5–13  
Filter TCP Access Permissions  
Window*



The **Filter** command for TCP access permissions allows you to use wildcard strings to restrict which access permissions are downloaded and displayed. For example, if **Owner Type** is set to either **User** or **Group**, you can filter on the User ID or group name, respectively.

## Web Access Permissions

As with TCP access permissions, Web access requires three port numbers to specify the destination, client, and server proxy connections. However, the destination port defaults to 80 and is embedded in the URL, and the client port defaults to 2080 and is specified during system configuration. This leaves only the server port to be specified.

The Web Access Table (Figure 5–14) displays the access records of all Web access permissions. Each Web access record contains either four or five fields, depending on whether or not a group is being assigned.

Owner Type	Owner ID	Perm. Type	URL / Group	Server Port
Group	all	Path	www.v-one.com/	2080
Group	all	Path	www.v-one.com/bacty	2080
Group	all	Path	www.v-one.com/betty	2080
Group	all	Path	www.sun.com/	2080
Group	Sales	Path	www.microsoft.com/	2080
Group	Sales	Path	www.sun.com/	2080
Group	Sales	Path	www.v-one.sales.com/	2080
Group	Finance	Path	www.v-one.finance.com/	2080
Group	Management	Include	Finance	
Group	Management	Include	Sales	
User	Shai21663	Path	www.v-one.com/	2080

**Figure 5-14**  
**Web Access Table**

**NOTE:** The Web Access permissions table reflects information found in the `sweb.acl` file located in the SmartGate Server’s root directory/etc on a UNIX-based server and SmartGate Server’s root directory\data on Windows NT.

- Owner Type** Specifies whether this permission is assigned to a user or a group.
- Owner ID** Either the User ID, the name or the group that will receive this permission, or “all.” All users belong to the universal group “all.”
- Permission Type** The value for this field can be either *Path* or *Include*. *Path* indicates that an individual path to a remote host and access permission has been designated. *Include* indicates that the specified Owner (either User or Group) will acquire the paths of the included group. Included group permissions display in green, while individual path permissions display in black.
- URL/Group** The Universal Resource Locator (URL) of the Web page or the name of the included group.
- Server Port** The SmartGate Server proxy port. The server port defaults to 2080 for Web (HTTP) access permissions. The Server Port is only used for individual paths.

**NOTE:** Wildcarded IP addresses, DNS names, and destination ports are valid. For more information, see Appendix C, “ACL Wildcarding.”

When managing Web access permissions, four commands are available:

- Add an access permission
- Edit an access permission
- Delete access permissions
- Filter display of access permission records

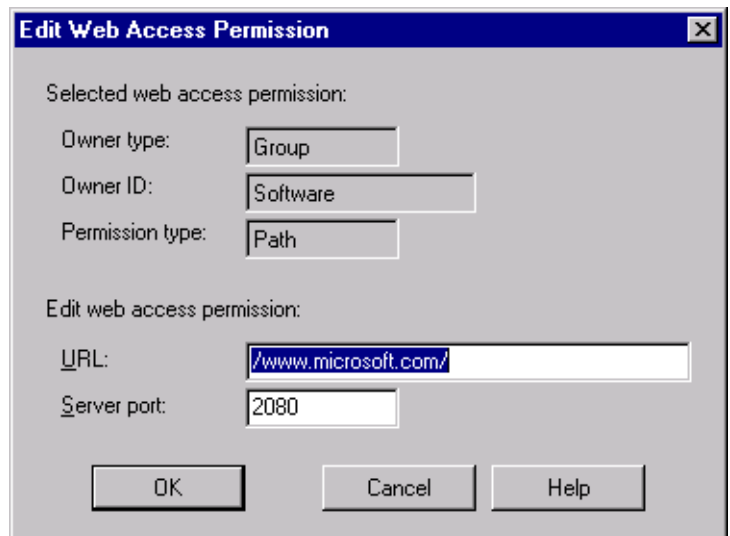
### Add/Edit Web Access Permissions

To add or edit Web access permissions, use either the buttons at the bottom of the window or the menu bar options. Figure 5-15 is an example of the Edit Web Access Permission Window. The Add Web Access Permission Window contains the same information in a nearly identical layout.

**Figure 5-15**  
*Edit Web Access Permission Window*

**NOTE:** There must be a “/” at the beginning of the URL and at the end of the destination (host domain and optional port).

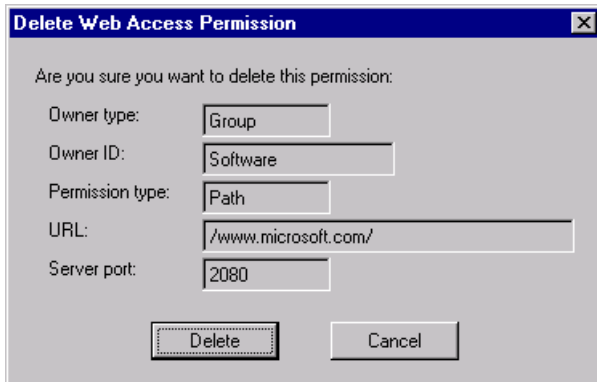
**NOTE:** URLs can start with “http://”.



The Add/Edit Web Access Permission Windows are self-explanatory—using the field descriptions on the preceding page—fill in the blanks with the appropriate information.

### Delete Web Access Permissions

To delete a Web access permission, select an access permission record or multiple records and either click **Delete** at the bottom of the window or use the menu bar option. Figure 5-16 is displayed.

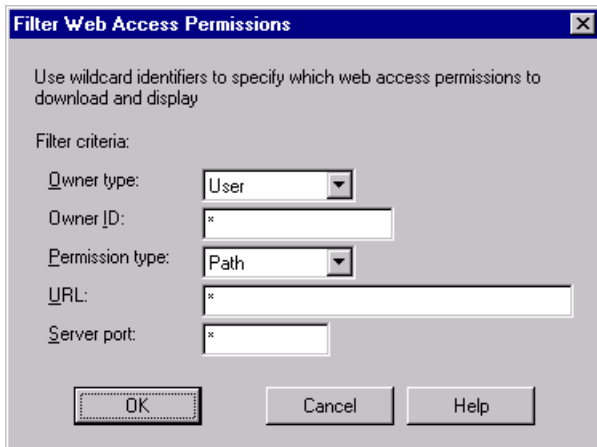


**Figure 5-16**  
**Delete Web Access Permission**  
**Window**

The Delete Web Access Permission Window is a confirmation that you want to delete the permission. If you select multiple permissions for deletion, the system will ask for a deletion confirmation on each permission.

## Filtering

To display specific Web access permission records, select **View** in the menu bar, and then select **Filter Web Access Permissions**. Figure 5-17 is displayed.



**Figure 5-17**  
**Filter Web Access Permissions**  
**Window**

The **Filter** command for Web access permissions allows you to use wildcard strings to restrict which access permissions are downloaded and displayed. For example, if **Owner Type** is set to either **User** or **Group**, you can filter on the User ID or group name, respectively.

## Setting Up On-Line Registration

The OLR Setup Table (Figure 5–18) allows you to manipulate the format of the data entry fields displayed to end users when they perform OLR.

**Figure 5–18**  
**OLR Setup Table**

**NOTE:** The OLR Setup Table reflects information found in the `reginfo.dat` file located in the SmartGate Server's root directory/etc on a UNIX-based server and SmartGate Server's root directory\data on Windows NT.

**NOTE:** Please refer to the *SmartGate With Netrust Authentication Guide* for more information on Netrust configuration options.

Field	Name	Length	Type	Arguments
VONE 1	First Name	20	Alphanumeric	
VONE 2	Last Name	20	Alphanumeric	
VONE 3	Social Security Number	9	SSN	
VONE 4	Group	30	Group List	Sales;Marketing;Documentation;Administration
ENTRUST 1	First Name	20	Alphanumeric	
ENTRUST 2	Last Name	20	Alphanumeric	

You can enter up to 10 data entry fields for each OLR method (VONE, ENTRUST, NETRUST, PAGER, or PKI), as follows:

**Fields 1 and 2 (Required)** When using the standard VONE OLR method, these fields are labeled, “First Name” and “Last Name” by default, but they can be changed. However, they must be alphanumeric without spaces or special characters. The combination of these two fields are used for different functions:

1. The first four characters of Field 1 are appended to 5 randomly generated numerals to create a 9 character User ID (unless you are using the UID Server).
2. Together, Fields 1 and 2 make up the user's long name in the user database. There is a space inserted between Fields 1 and 2.

Label these entry fields with titles that will be easy for your end users to understand. If you want to capture the full name of your end user, retain the default values.

**Fields 3 through 40** Use these fields to request any additional information you want to obtain from each end user. You may utilize as many or as few of these additional fields as you want, however, any fields that have been created must be answered by the end user during OLR. The format for each line is:

- 1. **Field** Defines the **OLR method**—V-ONE, ENTRUST, NETRUST, PAGER, or PKI—and the order in which the fields will appear during OLR.
- 2. **Name** Defines the title of the input field that the user sees during OLR (e.g., **Phone No.**).
- 3. **Length** Defines the maximum number of characters that the end user can enter in the field.
- 4. **Type** Defines what type of character(s) can be entered in the field. Select a type from the drop-down list.

Type	Syntax
Alphanumeric	alphabetic, 0–9
Number	0–9
Phone	0–9, dash/hyphen (-)
Group List	alphabetic, 0–9
Credit Card	0–9, dash/hyphen (-)
Passnum	0–9
Password	alphabetic
SSN	0–9, dash/hyphen (-)
Anything	no check

- 5. **Arguments** Defines a drop-down list of groups from which the user may choose during OLR. Applies only when type is **Group List**.

### OLR Branding Options

The Web page produced for OLR by the SmartGate Server may be branded by adding the following information to the OLR Branding Options Window (Figure 5–19). Click the **OLR Branding** button at the bottom of the OLR Setup Table.

**Figure 5-19**  
**OLR Branding Settings Window**

**NOTE:** The OLR branding information is found in the `sgconf.ini` file located in the SmartGate Server's root directory/etc on a UNIX-based server and SmartGate Server's root directory\data on Windows NT.

**NOTE:** When changing the OLR branding on the SmartGate Server, an administrator may **not** enter an "\*" because of wildcard matching.

OLR Branding Settings

Enter new OLR branding settings:

Startup description: SmartPass

Startup arguments: -h http://www.v-one.com/

Company name: V-ONE Corporation

Web page: www.v-one.com

Street address: 20250 Century Boulevard, Suite 300

City: Germantown

State: Maryland

Zip code: 20874

Country: USA

Phone number: (301) 515-5200

E-mail: pcleveland@v-one.com

All outside firewall: ☒

OK Cancel Help

If you enter the appropriate information into the **Startup description** and **Startup arguments** text boxes, a desktop icon will be produced by SmartPass which will start the SmartPass software and the default browser after the user has performed OLR.

1. **Startup description** Enter the title you want to appear under the SmartPass icon that will be placed on your end user's desktop after they register.
2. **Startup arguments** Enter the command **-h** plus the URL of the Web page you want the end user to see when they start SmartPass using their desktop icon. For example:  
**-h http://www.v-one.com/**
3. **Company name through e-mail** Enter your company address information.
4. **All outside firewall** Select this check box, only if none of your users need to navigate a firewall to reach the Web and perform OLR.

# Assigning Administrative Rights

To configure the SmartGate administrator's rights, click the **Admin Rights** tab on the SmartAdmin window. Figure 5-20 is displayed.

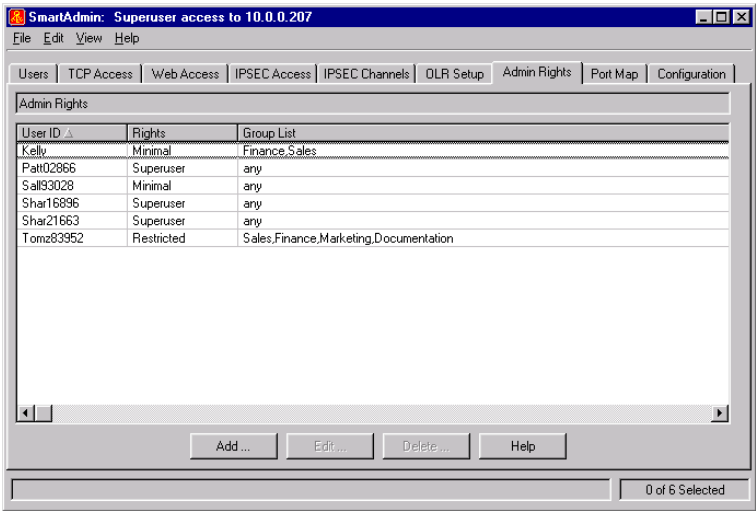


Figure 5-20  
Administrative Rights Table

**NOTE:** The Administrative Rights Table reflects information found in the `adm-gw.ac1` file located in the SmartGate Server's root directory/etc on a UNIX-based server and SmartGate Server's root directory\data on Windows NT.

The Administrative Rights Table provides the means to assign administrative privileges to individual users. Click **Add** and Figure 5-21 is displayed.

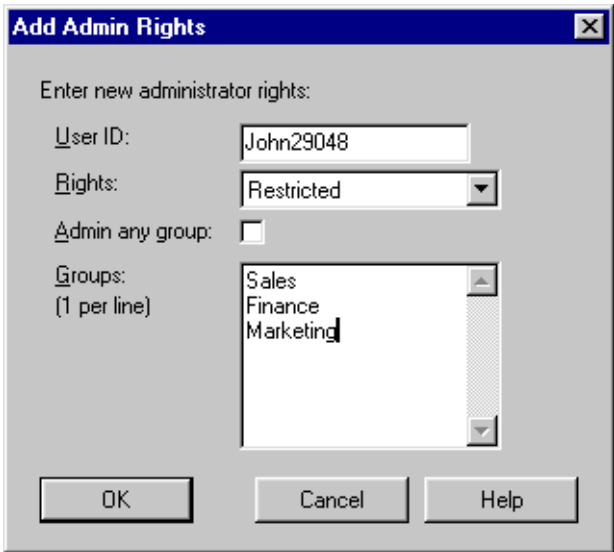


Figure 5-21  
Add Administrative Rights Window



**NOTE:** A User ID in the `adm-gw.acl` file that is not configured fully will not have SmartAdmin administrative access.

**NOTE:** Keep in mind when giving administrator rights to specific groups, that any users who were not assigned a group during OLR are defaulted to the **gp** group.

Enter the following information for each user that you want to assign administrative privileges.

1. The administrator's User ID.
2. The administrative rights:
  - **Minimal** Administrators at this level can only enable/disable users and edit a user's name in the event of a name change or a typographical error. Access at this level may be limited to certain groups.
  - **Restricted** Administrators at this level have full access to user data. In addition to those rights provided at the minimal level, administrators can change authentication keys and add/edit/delete end users. Access at this level may be limited to certain groups
  - **Standard** Administrators at this level have full access to access permissions. In addition to those rights provided at the restricted level, administrators can add/edit/delete access permissions. Access at this level may be limited to certain groups.
  - **Superuser** Administrators at this level have access to all settings. In addition to the privileges of the standard administrator, superusers can assign administrator levels, change SmartGate configuration settings, and configure the OLR, IPSec Channels, PKI, and Single Port Proxy Map files. Superusers have full access to all groups.
3. The groups to which the administrator has rights, or select the **Admin any group** check box for all groups.

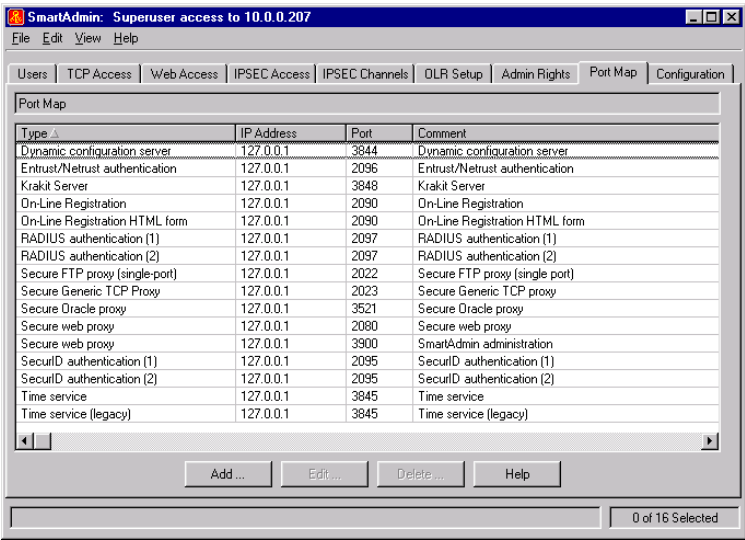
## Single Port Client

When SmartPass is started, it obtains a list of SmartGate Servers from the user's authentication token. Then, for each server, it connects to the time service and obtains the server's time. Next, it performs a secure dynamic configuration with the server by downloading the user's current access permissions. If a SmartGate Server is operating in single-port mode, then it will indicate this in the response returned from the time service. SmartPass can communicate with a mixture of single-port and "legacy" SmartGate Servers simultaneously.

# Configuring Single Port Proxy

The SmartGate Single Port Proxy provides the various SmartGate services with a single-port presence on the perimeter of a network. This means that all SmartPass to SmartGate (client to server) connectivity will pass through the Single Port Proxy and be forwarded to the correct destination SmartGate service.

The Single Port Proxy must determine the SmartGate service destination for each SmartPass connection arriving on its single port. To do this, it uses a port mapping file, `sgproxy.conf`. A default set of rules is installed with the SmartGate Server software (Figure 5–22).



Type	IP Address	Port	Comment
Dynamic configuration server	127.0.0.1	3844	Dynamic configuration server
Entrust/Netrust authentication	127.0.0.1	2096	Entrust/Netrust authentication
Krakit Server	127.0.0.1	3848	Krakit Server
On-Line Registration	127.0.0.1	2090	On-Line Registration
On-Line Registration HTML form	127.0.0.1	2090	On-Line Registration HTML form
RADIUS authentication (1)	127.0.0.1	2097	RADIUS authentication (1)
RADIUS authentication (2)	127.0.0.1	2097	RADIUS authentication (2)
Secure FTP proxy (single-port)	127.0.0.1	2022	Secure FTP proxy (single port)
Secure Generic TCP Proxy	127.0.0.1	2023	Secure Generic TCP proxy
Secure Oracle proxy	127.0.0.1	3521	Secure Oracle proxy
Secure web proxy	127.0.0.1	2080	Secure web proxy
Secure web proxy	127.0.0.1	3900	SmartAdmin administration
SecurID authentication (1)	127.0.0.1	2095	SecurID authentication (1)
SecurID authentication (2)	127.0.0.1	2095	SecurID authentication (2)
Time service	127.0.0.1	3845	Time service
Time service (legacy)	127.0.0.1	3845	Time service (legacy)

SmartGate administrators can adjust this table according to their network configuration. Table 5–2 lists the default Single Port Proxies.

**NOTE:** The default for the Single Port Proxy is 3845.

**NOTE:** The `sgproxy.conf` file is located in the SmartGate Server’s root directory/ etc on a UNIX-based server and SmartGate Server’s root directory\data on Windows NT.

*Figure 5–22  
Single Port Proxy Map Table*

**NOTE:** All of the single port mapping rules are loaded during the installation process.

**Table 5-2**  
**Default Single Port Proxies**

Type	Default	Comment
Dynamic Configuration	3844	Dynamic configuration
Entrust/Netrust Authentication	2096	Entrust/Netrust authentication
Krakit Server	3848	Krakit Server
On-Line Registration	2090	On-Line registration
On-Line Registration HTML	2090	On-Line registration HTML form
RADIUS Authentication (1)	2097	RADIUS authentication
RADIUS Authentication (2)	2097	RADIUS authentication
Secure FTP Proxy (single-port)	2022	Secure FTP proxy single-threaded
Secure Web Proxy	2080	Secure Web proxy
Secure Web Proxy	3900	SmartAdmin Administration
RSA SecurID Authentication (1)	2095	RSA SecurID authentication
RSA SecurID Authentication (2)	2095	RSA SecurID authentication
Secure Oracle Proxy	3521	Secure Oracle proxy
Secure TCP	2023	Secure generic TCP proxy
Time Service	3845	Time service provides standard handling of the SmartGate time service
Time Service (legacy)	3845	Time service (legacy) provides backward compatibility for time server requests from SmartPass clients prior to 3.1

## Add/Edit Port Map Rules

The Port Map Window is used to configure Single Port Proxy paths (Figure 5-23).

**Figure 5-23**  
**Add Port Map Window**

Enter new port map:

Type: Secure Generic TCP Proxy

IP address:

Port:

Comment:

OK Cancel Help

1. **Type** Select from the drop-down box the type of service.
2. **IP Address** Type the IP address of the service.
3. **Port** Type the port number.
4. **Comment** Type any information pertaining to that rule.

## Changing the Default Single Port Proxy

The SmartGate System is configured to use port 3845 as the default Single Port Proxy. If you are changing the default Single Port Proxy, you must configure **both** the SmartGate Server and the SmartPass software.

### SmartGate Server

#### All platforms:

- Using SmartAdmin, change both Time Service single port mapping rules to the new number.

#### Microsoft Windows NT:

- Click the **Start** button and select **Run**. Type **regedit** and click **OK**. The Windows Registry Editor will be displayed. Select:

**HKEY\_LOCAL\_MACHINE\SOFTWARE\V-ONE\Smartgate\4.x\extensions\sgproxy\port**

Double-click **port** and enter the new port value.

#### UNIX-based:

- There are two files that should be edited using a UNIX editor, such as vi or pico.

Open the `services` file located in the `/etc` directory. Replace 3845 (the default) in the following lines with the new port number:

```
sgtmsrv 3845/tcp
sgproxy 3845/tcp
```

Open your operating system's SmartGate start-up file using the following locations:

BSD/OS:	<code>/etc/rc.local</code>
Solaris:	<code>/etc/rc2.d/S90sgate</code>
RedHat Linux:	<code>/etc/rc.d/rc3.d/S90sgate</code>

Replace 3845 (the default) in the following lines with the new port number:

```
/usr/smartgate/libexec/sgproxy -daemon 3845
```

**NOTE:** The option of changing the Single Port Proxy default is only available with SmartPass 3.3 and later.

**NOTE:** The registry keys can be displayed in either decimal or HEX format.

**NOTE:** The changes you make to the start-up file for SmartGate **WILL NOT** be saved during an upgrade of the SmartGate Server software.

**Macintosh USERS:** To change the default Single Port Proxy on SmartPass, see "Changing the Default Single Port Proxy on SmartPass for the Macintosh" in Chapter 5, "SmartPass for the Macintosh," of the *SmartPass Administrator's Guide*.

**NOTE:** xxxxxx can be any name you choose.

## SmartPass

There are two steps involved in changing the Single Port Proxy default on the SmartPass software: a new registry key must be included on the SmartPass installation disk by the SmartGate administrator; and then it must be imported into the registry by the end user:

- After installing SmartPass, open the Windows Registry Editor (run **regedit**) to:

**HKEY\_CURRENT\_USER\Software\V-ONE\smartpass\4.x\options**

If the **Time port** key exists (right side of the Windows Registry Editor window) double-click to edit the value. Click the decimal radio button on the right and enter the new port value in the window to the left.

If the **Time port** key does not exist, click **edit** on the toolbar of the Windows Registry Editor. From the **edit** pull-down menu, select **new** and then **DWORD Value**. A new key will appear in the right window and the name field will be highlighted. Type in **Time port** and click **ENTER**. Double-click the key and enter the new port value in the window to the left.

You must then export this registry key to your SmartPass Installation disk. It will appear on the disk as xxxxxx.reg.

OR

Using a text editor, such as NotePad, the SmartGate administrator can create a file named xxxxxx.reg, with the following information, and include it on the SmartPass installation disk:

```
REGEDIT4
```

```
[HKEY_CURRENT_USER\Software\V-ONE\smartpass\4.x\options  
"Time port"=dword:0000f05
```

where: 0000f05 is the port number in HEX format.

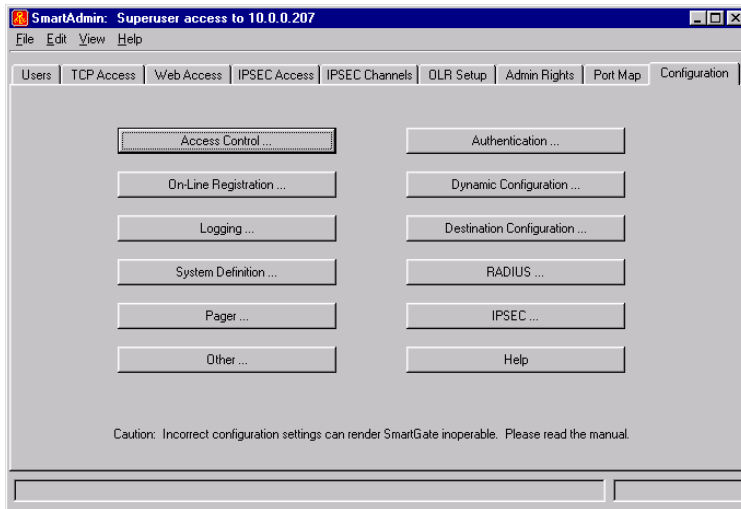
- After the end user has installed the SmartPass software, he must double-click the xxxxxx.reg file. This will automatically update his registry key:

```
HKEY_CURRENT_USER\Software\V-ONE\smartpass\4.x\  
options\Time port
```

with the new port number.

## Setting Configuration Options

To configure the SmartGate Server, click the **Configuration** tab on the SmartAdmin window. Figure 5–24 is displayed.



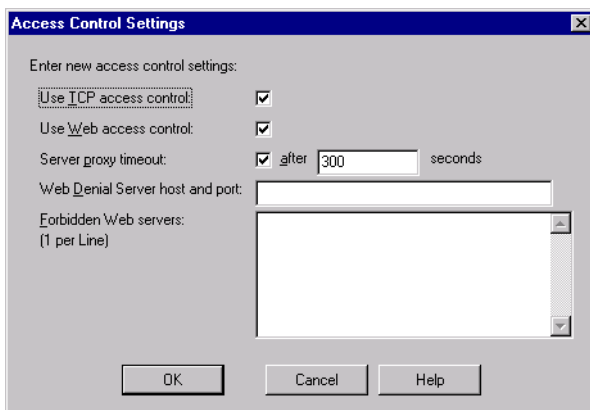
**Figure 5–24**  
**Configuration Window**

**NOTE:** The Configuration Window reflects information found in the `sgconf.ini` file located in the SmartGate Server's root directory/ etc on a UNIX-based server and SmartGate Server's root directory\data on Windows NT.

The Configuration Window provides SmartGate administrators with the means to set the SmartGate Server's configuration variables. SmartAdmin groups these variables into categories.

## Access Control Settings

To open the Access Control Settings Window (Figure 5–25), click **Access Control** on the Configuration Window.



**Figure 5–25**  
**Access Control Settings Window**

**NOTE:** The file names in parentheses are the actual name values in the `sgconf.ini` file.

### Use TCP access control (`sgateacl`)

Specifies whether SmartGate will be used to grant or deny users' permissions to access services through the generic (`sgate`), FTP (`sgftp`), or Oracle (`sgora`) proxies.

1. If the **Use TCP access control** check box is selected, access control is turned **on**. After successful user authentication, access to all TCP services is controlled through SmartGate's Authentication Server.

Access control is turned on by default. It is strongly recommended that you leave it on.

If access control is selected, you must use Dynamic Configuration rather than Manual Setup for setting secure pathways.

2. If the **Use TCP access control** check box is not selected, access control is turned **off**. All requested client connections will be allowed after successful authentication.

### Use Web access control (`swebacl`)

Specifies whether SmartGate will be used to grant or deny users' permissions to access Web services.

1. If the **Use Web access control** check box is selected, access control is turned **on**. After successful user authentication, access to the Web service is controlled through SmartGate.

Access control is turned on by default. It is strongly recommended that you leave it on.

If access control is selected, you must use Dynamic Configuration rather than Manual Setup for setting secure pathways.

2. If the **Use Web access control** check box is not selected, access control is turned **off**. All requested Web connections will be allowed after successful authentication.

### Server proxy timeout (`max_quiet_time`)

Specifies the number of seconds of idle time before a SmartGate Server (proxy) will timeout and close its connection to SmartPass. You can enter a setting of up to 8 digits. Deselect the **Server proxy timeout** check box to indicate no timeout. The default setting is 300 seconds (5 minutes).

### Web Denial Server host and port (`denial_server`)

Specifies either the hostname or IP address and the port number of a customer-written Denial Server.

This Denial Server follows a preset protocol to report back to SmartGate on whether a Web access request should be granted or denied. Because the Denial Server is checked before the Web Proxy (`sweb`), it is capable of overriding those permissions. The Denial Server listens on a specified port, receives details on Web requests, and returns a response of “GRANT,” “DENY,” or “PASS,” which it then passes on for checking against the permissions assigned in `sweb.acl`. This feature is optional. If not designated there will be no Denial Server.

#### **Format: *host:port***

If you enter a port but no host, *host* will default to `localhost` (`127.0.0.1`).

### Forbidden Web servers (`sweb_not_allowed`)

Specifies a list of Web servers to which the Web Proxy (`sweb`) is not allowed to connect. You can enter either hostnames or IP addresses. If you do not enter any Web servers in this box, all Web servers will be allowed.

**NOTE:** See “[Denial Server](#)” in Appendix B, “Services,” for more information.



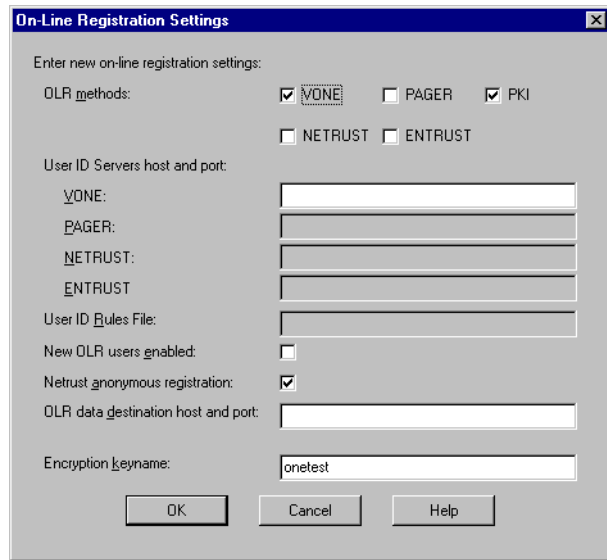
**Figure 5–26**  
**On-Line Registration Settings**  
**Window**

**NOTE:** Please refer to the *SmartGate With Netrust Authentication Guide* for more information on Netrust configuration options.

**WARNING!** All programs that use the user database (i.e., the Dynamic Configuration Server, UID Server, KRAKit Server, etc.) must reside on the same computer where the user database is stored (i.e., the Authentication Server).

## On-Line Registration Settings

To open the On-Line Registration Settings Window (Figure 5–26), click **On-Line Registration** on the Configuration Window.

The image shows a Windows-style dialog box titled "On-Line Registration Settings". It contains several configuration options. Under "Enter new on-line registration settings:", there are checkboxes for "VONE", "PAGER", "PKI", "NETRUST", and "ENTRUST". "VONE" and "PKI" are checked. Below this, there are text input fields for "User ID Servers host and port:" with sub-labels "VONE:", "PAGER:", "NETRUST:", and "ENTRUST:". There is also a field for "User ID Rules File:". A checkbox for "New OLR users enabled:" is unchecked. A checkbox for "Netrust anonymous registration:" is checked. A text input field for "OLR data destination host and port:" is empty. At the bottom, there is a text input field for "Encryption keyname:" containing the text "onetest". At the very bottom are three buttons: "OK", "Cancel", and "Help".

### OLR methods (OLRMethod)

This setting specifies the possible methods—V-ONE, NETRUST, ENTRUST, PAGER, or PKI—used to add users during OLR. Select all of the methods that apply. V-ONE is the standard default method and should remain selected.

The **OLR methods** option is used in conjunction with the **Netrust anonymous registration** and the **User ID Servers host and port** options, and in addition to the **OLR Setup** tab, to specify Netrust authentication using a UID Server.

### User ID Servers host and port (uid\_server)

Specifies either the hostname or IP address and the port number of the optional **SmartGate User ID (UID) Servers** (sguidsrv). Up to four separate UID Servers can be used depending on the authentication method. This setting is used in conjunction with the **User ID Rules File** setting.

The UID Server is sent the user's OLR information and must return either a User ID to be assigned to the user or a reason for rejection.

**Format: *host:port***

The port number defaults to 3846 and should not be changed.

There is no system default; if a UID Server is not specified, the feature will not be used.

There are three possible scenarios:

**Scenario 1:** No UID Server. SmartGate will automatically generate a User ID. This is the SmartGate default.

**Scenario 2:** Use V-ONE's UID Server. SmartGate will generate a User ID according to the parameters you have set in the Rules File. The following are the procedures that need to be taken to use this option.

- Create a Rules File. See [“Creating a Rules File”](#) in Chapter 7, “On-Line Registration Services.”
- Specify the location of the Rules File in the **User ID Rules File** setting.
- Specify the **User ID Servers host and port** number. If the UID Server resides on the same computer as your SmartGate Server, set this to localhost (127.0.0.1).
- **UNIX-based:**

Edit the `/etc/inetd.conf` file and uncomment the `sguidsrv` line:

```
sguidsrv stream tcp nowait root /usr/smartgate/  
libexec/sguidsrv sguidsrv
```

Restart `inetd` by rebooting or signal `inetd` by typing:

Solaris:

```
ps -ef|grep inetd  
kill -HUP process_id
```

BSD/OS:

```
ps -ax|grep inetd  
kill -HUP process_id
```

where: the *process\_id* number for `inetd` is returned by the system after typing the first command as described above.

**NOTE:** To access these options using SmartAdmin, select the **Configuration** tab, and then the **On-Line Registration** button.

**NOTE:** If you must change the default port number of the SmartGate UID Server, see [“UID Server for On-Line Registration”](#) in Chapter 7 “On-Line Registration Services” for detailed instructions.

**NOTE:** The **New OLR users enabled** setting does not apply to RADIUS, RSA SecurID, and Netrust (when anonymous registration is allowed) authentication.

**NOTE:** Depending on your security policy, this may be a security risk.

**NOTE:** Please refer to the *SmartGate With Netrust Authentication Guide* for detailed information on SmartGate configuration options.

## ■ Microsoft Windows NT:

The UID Server (sguidsrv) is running by default—no further setup is required. See “[Adding and Removing Services](#)” in Chapter 4, “Installing the SmartGate Server Software - Windows NT,” for more information on services.

### Scenario 3: Use a Custom UID Server.

- Specify a UID Server (*host:port*) in the **User ID Servers host and port** setting to go to your process.
- Create your own UID Server process and run it. See “[Creating Your Own UID Server Process](#)” in Chapter 7, “On-Line Registration Services.”

## User ID Rules File (UidFile)

Specifies the location (full path name) of the [Rules File](#) for use with the optional UID Server. This setting is used in conjunction with the **User ID Servers host and port** setting.

### Format: *location*

where: *location* is the name and full path name of the Rules File.

See “[Creating a Rules File](#)” in Chapter 7, “On-Line Registration Services,” for a description on how to create a Rules File.

## New OLR users enabled (online\_reg\_enable)

Specifies whether you want your end users to be enabled immediately after performing OLR or not. If you do not, you must enable them manually, either through remote administration or by using the command line `sgadm`.

SmartAdmin can be used to manually enable users by clicking the **Users** tab, double-clicking on a specific user record, and selecting the **Enable** check box.

## Netrust anonymous registration (anon\_reg\_allowed)

Specifies if SmartGate allows for anonymous registration when using the Netrust authentication method. When using Netrust authentication, the SmartGate administrators have two options. They can allow their end users to register anonymously, the default; or they can have them perform OLR, in which case they must disallow anonymous registration and set up a UID Server.

The **Netrust anonymous registration** option is used in conjunction with the **OLR methods** and the **User ID Servers host and port** options, and in addition to the **OLR Setup** tab, to specify Netrust authentication using a UID Server.

### OLR data destination host and port (online\_reg\_service)

Specifies either the hostname or IP address and the port number of an OLR Activity Recording Service (ARS). The ARS specified will receive OLR activity information at the end of each OLR session. There is no default; the OLR Server will not send this information if you leave this field blank.

#### Format: *host:port*

The information that the user enters on the screen during OLR is stored locally in the `sgreg usr` file. At the end of each OLR session, the OLR Server generates a record for the user. It writes the record to `sgreg usr` and also sends the following data to the ARS that you provide.

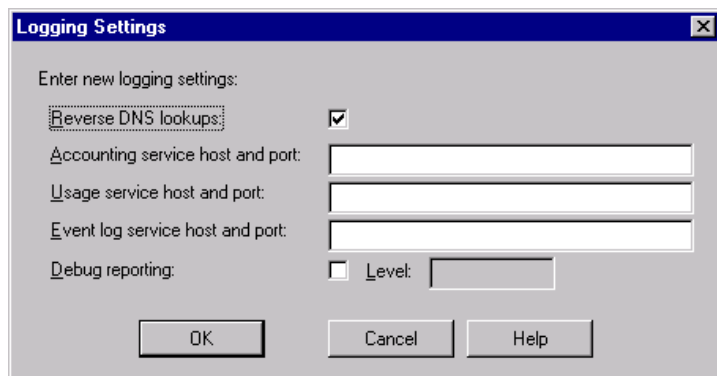
- Date and time the user was registered (local to the OLR Server).
- User ID and OLR data fields of the registered user.

### Encryption keyname (keyname)

Specifies the encryption keyname which is reflected in the certification files and subsequently certified by V-ONE Corporation.

## Logging

To open the Logging Settings Window (Figure 5-27), click **Logging** on the Configuration Window.



**NOTE:** You may specify any IP address and port number, although port number 3839 is recommended.

**NOTE:** The encryption keyname option is found in the `reginfo.dat` file (not `sgconf.ini`) in the SmartGate Server's root directory/ etc on a UNIX-based server and SmartGate Server's root directory\data on Windows NT.

**Figure 5-27**  
*Logging Settings Window*

**NOTE:** If **Reverse DNS lookups** is not checked only IP addresses (no hostnames) can be used in the **Authentication Server** and the **Configuration Server** settings.

**NOTE:** For more information regarding the Accounting Service, see “[Accounting Service](#)” in Appendix B, “Services.”

### **Reverse DNS lookups** (`dns_reverse`)

Specifies whether the server should attempt to do a reverse [Domain Name Service \(DNS\)](#) hostname lookup on all client connections.

1. If you select the **Reverse DNS lookups** check box, a lookup is attempted.
2. If you do not select the **Reverse DNS lookups** check box, a lookup is not attempted and the client IP address is written to `/var/log/smartgate` with a hostname of “unknown.”

### **Accounting service host and port** (`accounting_service`)

Specifies either the hostname or IP address and port number of the accounting service that will receive accounting information at the end of each SmartGate session. If this field is blank, the SmartGate Server will not send accounting information. The following information is sent:

- IP address of SmartPass
- ID of the user originating the session
- Session start time
- Session end time
- Destination host
- Destination port
- Server to client number of bytes sent
- Client to server number of bytes sent
- The word “misc”

**Format:** *host:port*

If only the host is entered, the port number defaults to 4839; this is recommended, but you may enter a different port number if needed.

### **Usage service host and port** (`stat_server`)

Specifies either the hostname or IP address and port number of the statistics service that the SmartGate Server will send session start and end log entries to. They can be used to create statistics and reports on user sessions.

**Format:** *host:port*

If this field is blank, the SmartGate Server will not send statistics information.

## Event log service host and port (event\_log)

Specifies either the hostname or IP address and port number of an optional UDP Server. If used, the Event Log will be sent the same log entries as the flat file in the SmartGate Server's root directory.

### Format: *host:port*

If this field is blank, the SmartGate Server will not send event log information.

## Debug reporting (debug)

Specifies what level, if any, of debug information will be written, in addition to standard logging information, to the Microsoft Windows NT Event Viewer or to `/var/log/smartgate` on a UNIX-based SmartGate Server. The primary purpose of this setting is to obtain additional log messages about the operations of the SmartGate Server.

1. If you do not want debug information, deselect the **Debug reporting** check box, which is the default setting and only standard error messages and user connection information is written to the log. Debug level "0" is equivalent to non-debug, standard logging.
2. If you want debug information, select the **Debug reporting** check box and then type a "1" or higher in the **Level** box to indicate the level of debug reporting.
3. If you want to lower your logging information to report only error messages to the log, select the **Debug reporting** check box and type a "-1" in the **Level** box.

**NOTE:** The higher the debug level, the more information is logged. However, performance may be affected.

**Figure 5–28**  
**System Definition Settings**  
**Window**

**WARNING!** The **SmartGate Server name** is required! You must ensure that the appropriate value is present. Do not change this value or you will be locked out of SmartAdmin.

**WARNING!** To use IPsec features or UDP broadcasting (UDPPortList), an IP address—NOT a hostname—must be used for the Server name.

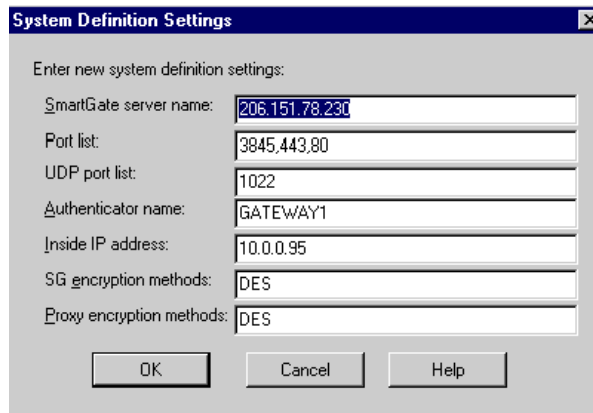
**WARNING!** The **Authenticator name** is required! You must ensure that the appropriate value is present. Do not change this value or you will be locked out of SmartAdmin.

**WARNING!** You must enter the **Inside IP address**.

**NOTE:** InsideIP accepts a comma separated list. The server will broadcast with each of the IP addresses (together with the UDPPort) to all interfaces and will then wait for the return packet.

## System Definition Settings

To open the System Definition Settings Window (Figure 5–28), click **System Definition** on the Configuration Window.



### SmartGate Server name (domainname)

Specifies the hostname or IP address of your SmartGate Server (specifically, where your OLR Server resides).

### Port list (PortList)

This setting allows the SmartGate Server administrator to set up the server to listen on multiple ports. This allows end users to try to connect using ports other than the standard single port proxy port 3845 to navigate through firewalls.

### UDP port list (UDPPortList)

This setting allows the SmartGate Server administrator to set up the server to look up a UDP port list on an sgate service. In order for the Citrix ICA Client “Auto-Locate” feature to work, you must create a TCP Access for 255.255.255.255 Client port=1604 Server port=2023 Destination port=1604.

### Authenticator name (authenticator)

This value identifies the smart card **Authenticator name**. The Authenticator name is stored on the user’s physical or virtual smart card with the user’s authentication key. This value should be inserted automatically by SmartGate.

### Inside IP address (InsideIP)

This setting enables you to specify the IP address assigned to the inside network adapter card on the SmartGate Server.

## SG encryption methods (SSEncryptMethod)

This setting defines the complete set of encryption methods, either DES or 3DES, available to be used for OLR, Time Server, Dynamic Configuration, and Authentication. This setting is communicated once at the initial connection of a SmartGate session and remembered by the client for its duration. The value string is displayed as a comma-delineated list of available methods; the preferred method will be listed first. The default method is DES encryption.

**Format:** *3DES,DES*

## Proxy encryption methods (ProxyEncryptMethod)

This setting defines the complete set of encryption methods available to be used for proxy data packets which convey end user data between client and server. The possible values are 3DES, DES, and RC4. This setting is communicated once at the initial connection of a SmartGate session and remembered by the client for its duration. The value string is displayed as a comma-delineated list of available methods; the preferred method will be listed first. The default method is DES encryption.

**Format:** *3DES,DES,RC4*

**NOTE:** Triple DES (3DES) encryption is only available with SmartGate 2.6 and SmartPass 3.3 and later versions. If previous versions are used for either server or client, this setting will be ignored.

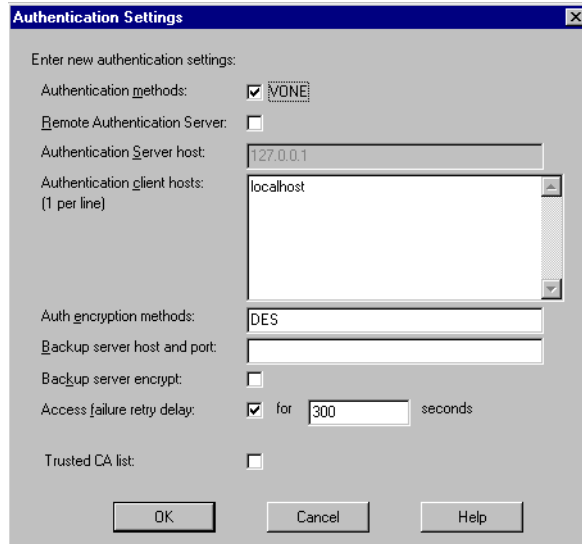


**Figure 5–29**  
**Authentication Settings Window**

**WARNING!** All programs that access the user database (i.e., the Dynamic Configuration Server, UID Server, KRAKit Server, etc.) must reside on the same computer where the user database is stored (i.e., the Authentication Server).

## Authentication Settings

To open the Authentication Settings Window (Figure 5–29), click **Authentication** on the Configuration Window.



### Authentication methods (AuthMethod)

This setting specifies the possible methods—VONE and PAGER—used to authenticate users. Select all of the methods that apply. VONE is the standard default method and should remain selected.

### Remote Authentication Server (sgasrv)

Select this check box if you are configuring your SmartGate Server to connect to a [Remote Authentication Server](#). If selected, the **Authentication Server host** setting, directly below, will become available.

### Authentication Server host

Type into the **Authentication Server host** text box either the hostname or IP address of the remote Authentication Server. This setting is used by the SmartGate proxies when they are not on the same host as the Authentication Server. Change this setting only if you have installed your Authentication Server on a computer other than the SmartGate Server. This option allows more than one SmartGate Server to share one Authentication Server.

If this option is not set, localhost (127.0.0.1) is used as the Authentication Server by default.

If you are using the **Remote Authentication Server** setting, you must also configure the `sgconf.ini` file on the computer where the Authentication Server resides to recognize the SmartGate Server(s) connecting to it. Use the **Authentication client hosts** setting on that computer.

### **Authentication client hosts** (`sgasrv_clients`)

This setting allows you to define a list of SmartGate Servers, in addition to `localhost`, from which the Authentication Server will allow connections. The Authentication Server only accepts requests from `localhost` and the hosts specified by this setting. The number of hosts you may specify is limited to the maximum line length (255). `localhost` is always assumed.

Use this option if the Authentication Server is being shared by other SmartGate Servers.

The **Authentication client hosts** setting is defined on the computer where the Authentication Server resides.

### **Authentication encryption methods** (`AuthEncryptMethod`)

This setting specifies the method in which SmartGate Authentication traffic will be encrypted. The authentication traffic includes all traffic between the Authentication Server (`sgasrv`) and its clients (`sgasrv_client`) and the Configuration Server (`sgccsrv`) and its clients (`sgccsrv_client`). Available methods include plain (no encryption), DES, and 3DES. DES is the default method.

### **Backup server host and port** (`backup_userdb`)

Use this setting to specify either the hostname or IP address and the port number of the backup host. All changes to the user database on the Authentication Server will be mirrored on the backup host.

**Format:** *host:port*

This feature is optional. Unless a backup host has been set up and this option is specified, no backup will be created.

**NOTE:** Only IP addresses (no hostnames) can be used in the **Authentication Server host** setting if **Reverse DNS lookups** is not selected.

**NOTE:** The Authentication Server always listens to port 3838 for requests from the SmartGate Server.

**NOTE:** The time difference between the SmartGate Server and the Authentication Server must not be greater than 5 minutes.

**NOTE:** Only IP addresses (no hostnames) can be used in the **Authentication client hosts** setting if **Reverse DNS lookups** is not selected.

**NOTE:** Triple DES encryption (3DES) is only available with SmartGate 2.6 and SmartPass 3.3 and later versions. If previous versions are used for either server or client, this setting will be ignored.

**WARNING!** Your backup server must be a client of your SmartGate Server. Do this by adding the hostname or IP address of your backup host to the **Authentication client hosts** setting, as described above.

**NOTE:** The maximum value for Access Failure Retry Delay is 99999999.

**NOTE:** The validity date check is to verify the “not before date” and the “not after date.”

### Access failure retry delay (RETRY\_DELAY)

Use this setting to specify the minimum number of seconds allowed between unsuccessful user login attempts. If this is checked, the user will be forced to wait the specified number of seconds before trying again after an unsuccessful attempt. If this is not set or set to zero, the user’s ID will be locked after three consecutive unsuccessful attempts, and he will need to reregister.

### Trust CA list (TrustedCAList)

Use this setting to specify the level of verification when using PKI authentication. If this value is set to **yes** and you have added CA certificates to the trusted CA list using the `certmanager` program, the SmartGate PKI authentication server will check the validity of the date of the user certificate, verify that the certificate is the same as the original OLR PKI certificate, and verify the user certificate by one of the trusted CAs in the list.

## Dynamic Configuration Settings

“Dynamic Configuration” is a SmartGate function that virtually eliminates the need for local configuration by the end user. The end user’s list of access permissions updates automatically every time SmartPass is started and at regular intervals as defined by the end user. Dynamic Configuration operates on the principle that the SmartGate Server is the final authority for service access or “fine-grain access control.”

Each time SmartPass is started, it prompts the user for his Access Code, which is required to access his available authentication keys (the 128-bit authentication keys entered as 32 hexadecimal characters). SmartPass contacts every SmartGate Server for which the user has an authentication key, and requests that user’s current list of access permissions, those services to which the user will be allowed access by each SmartGate Server. The combined list of available services for all SmartGate Servers for which there are authentication keys forms the basis for providing secure connections between the user’s workstation and the available SmartGate communities.

The programs involved in Dynamic Configuration are the Dynamic Configuration Server (`sgccsrv`) and the Dynamic Configuration agent (`sgccag`). The function of the configuration agent is to forward SmartPass’ configuration requests to the Dynamic Configuration Server, and return the server’s

response to SmartPass. The configuration agent resides on the SmartGate Server, but the Dynamic Configuration Server can be executed on a separate computer.

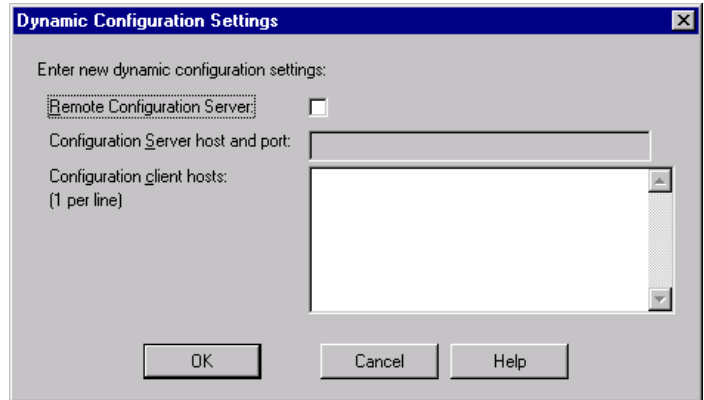
By default, the Dynamic Configuration Server resides on the same computer as the SmartGate Server. However, if you are distributing your SmartGate System across multiple processors, you may want more than one SmartGate Server to share one Dynamic Configuration Server. However, if the Dynamic Configuration Server is not on the same computer as the SmartGate Server, you must specify its location so that the configuration agent will be able to find it.

To open the Dynamic Configuration Settings Window (Figure 5-30), click **Dynamic Configuration** on the Configuration Window.

**Figure 5-30**  
**Dynamic Configuration Settings**  
**Window**

**WARNING!** All programs that access the user database (i.e., the Dynamic Configuration Server, User ID Server, KRAKit Server, etc.) must reside on the same computer where the user database is stored (i.e., the Authentication Server).

**NOTE:** Only IP addresses (no hostnames) can be used in the **Configuration Server host and port** setting if **Reverse DNS lookups** is not selected.



### **Remote Configuration Server (sgccsrv)**

Select this check box if you are configuring your SmartGate Server to connect to a remote Dynamic Configuration Server. If selected, the **Configuration Server host and port** setting, directly below, will become available.

### **Configuration Server host and port**

Type in the **Configuration Server host and port** text box either the hostname or IP address and the port number of the remote Dynamic Configuration Server. Use this setting only if you have installed your Dynamic Configuration Server on a computer other than the SmartGate Server.

The default of 127.0.0.1 (localhost) and port number 3843 are fixed values in the standard `sgconf.ini` file. Usually, you should retain these values. However, if you are distributing your SmartGate System across multiple processors, you will need to change them.

If you are using the **Remote Configuration Server** setting, you must also configure the `sgconf.ini` file on the computer where the Dynamic Configuration Server resides to recognize the SmartGate Server(s) connecting to it. Use the **Configuration client hosts** setting on that computer.

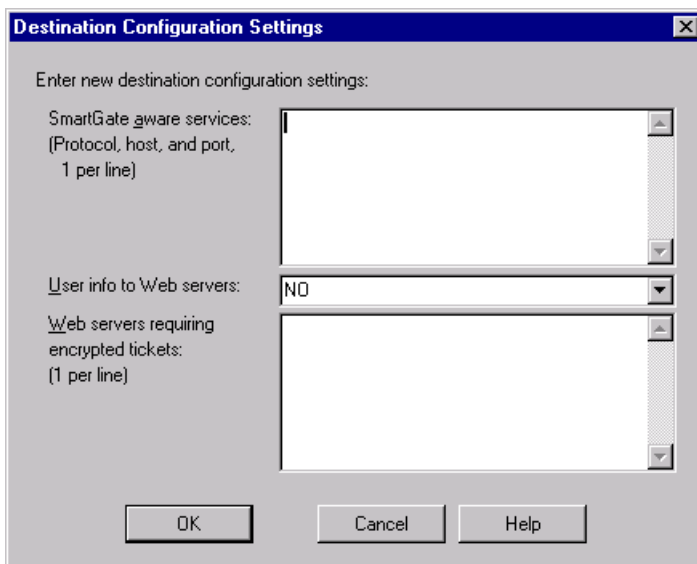
## Configuration client hosts (sgccsrv\_clients)

This setting allows you to define a list of SmartGate Servers, in addition to `localhost`, from which the Dynamic Configuration Server will allow connections. The Dynamic Configuration Server only accepts requests from `localhost` and the hosts specified by this setting. `localhost` is always assumed.

The **Configuration client hosts** setting is defined on the computer where the Dynamic Configuration Server resides.

## Destination Configuration Settings

To open the Destination Configuration Settings Window (Figure 5–31), click **Destination Configuration** on the Configuration Window.

The image shows a Windows-style dialog box titled "Destination Configuration Settings". It has a close button (X) in the top right corner. The main area contains three configuration sections. The first section is labeled "Enter new destination configuration settings:" and "SmartGate\_aware services: (Protocol, host, and port, 1 per line)". It features a large, empty text box with a vertical scrollbar. The second section is labeled "User info to Web servers:" and has a dropdown menu currently showing "NO". The third section is labeled "Web servers requiring encrypted tickets: (1 per line)" and features another large, empty text box with a vertical scrollbar. At the bottom of the dialog are three buttons: "OK", "Cancel", and "Help".

*Figure 5–31  
Destination Configuration  
Settings Window*

## SmartGate aware services (SmartGate\_aware)

Use this setting to specify a list of services that are ready to receive user information once that user has been authenticated by the SmartGate Server. The SmartGate Server will send SmartGate user's information only to those services specified in this list box.

**NOTE:** Only IP addresses (no hostnames) can be used in the **Configuration client hosts** setting if **Reverse DNS lookups** is not selected.

**Format:**    [*protocol*]:*host:port*  
              [*protocol*]:*host:port*

where: *protocol* is one of the following values:

<b>VONE:</b>	the default protocol
<b>VPOP:</b>	mail type of service
<b>nothing:</b>	defaults to VONE

*host* is the hostname or IP address of the SmartGate Aware Server.

*port* is the port number which the SmartGate Aware Server will be listening on.

### User info to Web server (UserInfoToWebServer)

Use this setting to specify whether the SmartGate user's information is sent to a SmartGate Aware Web server and, if so, the method and format in which to send the information. For any information to be sent to the Web server, it must be listed under **SmartGate aware services**. If this option is not specified, this feature is not used. The options in the drop-down box are:

**NO**    The SmartGate user's information will not be sent to the Web server.

**YES**    The SmartGate user's information will be appended to the URL, and placed in a separate Web message header line starting with **SMARTUSER**. The user's information can be retrieved by a CGI script from the environment variable **HTTP\_SMARTUSER**.

**MSG\_BODY** The SmartGate user's information will be sent to the Web server in the following format:

- appended to the URL.
- placed at the beginning of the body of the Web message, provided the message method is POST or PUT.

**name** The SmartGate user's information will be sent to the Web server in the following format:

- appended to the URL.
- placed in a separate Web message header line starting with *name*, a variable of your choice. In this way the environment variable becomes **HTTP\_name**.

**Formats:** The following are examples of the different ways that the user information will be formatted when sent to the Web server.

- If the user's information is appended to the URL:

```
?SMARTUSER=userid&SMARTGROUP=group&LONGNAME=longname&IP=IPaddress
```

- If the user's information is in a separate Web message header line:

```
SMARTUSER: userid&group&longname&IPaddress&
```

or

```
name: userid&group&longname&IPaddress&
```

- If the user's information is in the body part of a Web message:

```
?SMARTUSER=userid&SMARTGROUP=group&LONGNAME=longname&IP=IPaddress
```

where: *userid* is that user's ID.

*group* is the name of the group assigned to that user.

*longname* is that user's long name (the first 2 fields from the OLR Web page).

*IPaddress* is the IP address of that user's personal computer.

*name* is given above.

### Web servers requiring encrypted tickets

([ticket\\_to\\_web\\_server](#))

Use this option to specify the IP addresses or hostnames of those Web servers for which you want authentication performed using *sgkeys* as an encrypted ticket.

**NOTE:** Your Web server needs a CGI routine to decrypt the ticket and send a challenge back to the SmartGate Server.



**Figure 5-32**  
**RADIUS Settings Window**

**NOTE:** For more information on RADIUS authentication, see “Using RADIUS for User Authentication” in Chapter 6, “User Authentication.”

## RADIUS Settings

To open the RADIUS Settings Window (Figure 5-32), click **RADIUS** on the Configuration Window.

Host	Secret	Use CHAP	Wait

Time to live:

Challenge timeout:

OK Cancel Help

### RADIUS Backend Servers: Host (radius\_authsrv[1...5])

Specifies the hostname or IP address of the RADIUS Backend Server and its backup RADIUS Servers (will support up to 5 total). If RADIUS is being used, at least one value must be specified as the RADIUS Backend Server. Additional backup servers are optional.

### RADIUS Backend Servers: Secret (radius\_authsrv[1...5]\_secret)

Specifies the shared secrets for the RADIUS Backend Server and each of its backups (will support up to 5 total). If RADIUS is being used, at least one value must be specified for the RADIUS Backend Server. Additional backup servers are optional.

Each RADIUS Backend Server must be configured with its corresponding shared secret code. See your RADIUS documentation for further information.

### RADIUS Backend Servers: Use CHAP (radius\_authsrv[1...5]\_usechap)

Specifies if your RADIUS Backend Server is using CHAP authentication for its users. For each of these servers the SmartGate Server running the RADIUS module will simulate a CHAP exchange and send a CHAP-Password value instead of the normally hashed User-Password attribute. Each of the servers are defaulted to “no.”

## RADIUS Backend Servers: Wait

(radius\_authsrv[1...5]\_waitfor)

Set this to the number of seconds that you would like requests made to RADIUS to wait before they time out. Network factors may prevent certain servers from responding as quickly as they should. Defaults to 120 seconds for each server with a maximum of 32767 seconds. Do not use commas.

## Time to live (radius\_ttl)

Specifies the number of minutes for which the RADIUS authentication will remain valid before another authentication is required. The valid range is 1 to 1440 minutes and the default is 30 minutes.

## Challenge timeout (radius\_challenge\_timeout)

Specifies the number of minutes that a RADIUS challenge dialog box will remain on the screen before it times out. The valid range is 1 to 30 minutes and the default is 5 minutes.

## Other Settings

There are some miscellaneous options that do not fit into the defined categories. To open the Other Settings Window (Figure 5-33), click **Other** on the Configuration Window.

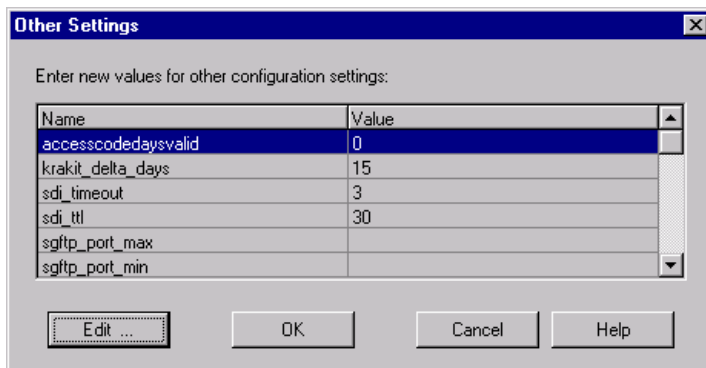


Figure 5-33  
Other Settings Window

## accesscodedaysvalid (AccessCodeDaysValid)

Use this setting to specify the maximum number of days that an Access Code can age before this server will require the user to change it. This value is downloaded to SmartPass during Dynamic Configuration and enforced by SmartPass. Any value between 1 and 999 days can be set, or it can be set to 0 (zero), in which case there is no maximum.

**NOTE:** For more information on KRAKit, see the standalone *KRAKit Guide*.

**NOTE:** For more information on SecurID authentication, see “Using RSA SecurID for User Authentication” in Chapter 6, “User Authentication.”

### **krakit\_delta\_days** (KraKit\_Delta\_Days)

This setting specifies the number of days that the KRAKit Server will be allowed to search for deleted user keys. The valid range is 1 to 1000 and the default is 15 days.

### **sdi\_timeout** (SDI\_TIMEOUT)

This setting specifies the number of minutes that the end user, when using RSA SecurID authentication, will be allowed before responding to a **Next Tokencode** or **New Pin Code** dialog box. The valid range is 1 to 30 minutes and the default is 3 minutes.

### **sdi\_ttl** (SDI\_TTL)

This setting specifies the number of minutes for which a RSA SecurID authentication will remain valid before another authentication is required. The valid range is 1 to 1440 minutes and the default is 30 minutes.

### **sgftp\_port\_max** (sgftp\_port\_max)

This setting specifies the maximum TCP port number to be used by the **FTP Proxy** for data transfers. This setting is used in conjunction with `sgftp_port_min` to limit the ports used by the FTP Proxy. The minimum port setting must be lower than the maximum port setting and the difference should be at least 10 ports.

### **sgftp\_port\_min** (sgftp\_port\_min)

This setting specifies the minimum TCP port number to be used by the FTP Proxy for data transfers. This setting is used in conjunction with `sgftp_port_max` to limit the ports used by the FTP Proxy. The minimum port setting must be lower than the maximum port setting and the difference should be at least 10 ports.

### **shim\_permitexe** (shim\_permitexe)

This setting specifies a list of executable files on the end user computer that will bypass the shim when accessing the Windows socket library (`wsock32`). If you are not using the shim, this setting is ignored.

**Format:** *executable1,executable2,executable3,...*

### **smartwebport** (SmartWebPort)

This setting specifies the port on which SmartPass should listen for Web connections. The default port is 2080.

# PKI Administration

The trusted PKI CA Server Certificates are listed under the PKI CA Tab in SmartAdmin, Figure 5–34.

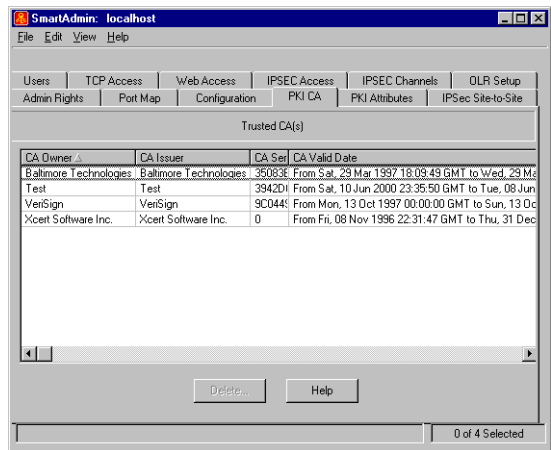


Figure 5–34  
PKI CA Window

Since the certificates must be installed via the command line, the only administration available on this tab is the ability to delete certificates. The delete function is only accessible to SmartGate SmartAdmin superusers.

The PKI Certificate Attributes are listed under the PKI Attributes Tab in SmartAdmin, Figure 5–35.

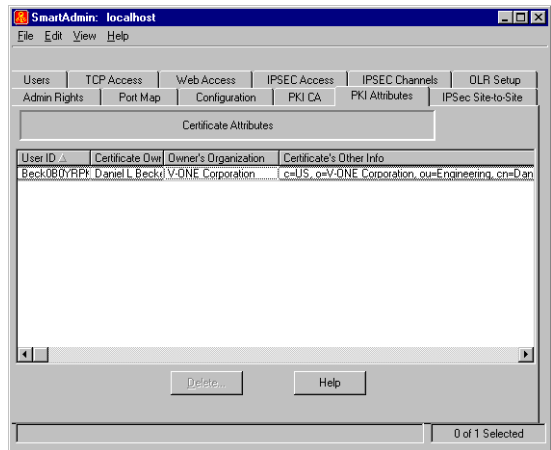


Figure 5–35  
PKI Attributes Window

All valid PKI certificates are listed here. The only administrative function available to SmartGate SmartAdmin superusers is the delete function.

**NOTE:** Instructions for adding CA Server certificates is located in [“Adding CA Certificates to the Trusted CA List,”](#) in Chapter 6, “User Authentication,” of this guide.

# Chapter 6

## User Authentication

### SmartGate Authentication Server

The SmartGate Authentication Server (*sgasrv*) is the final authority that determines whether any user attempting to connect via SmartGate will be granted access. It not only validates the identity of the user, but also decides whether that user's access request is legitimate.

In addition to user authentication, the Authentication Server also manages access control for all services made available to users of any given SmartGate-controlled community. Finally, the Authentication Server is a cooperating component of On-Line Registration (OLR) and is responsible for adding new users to the authentication database after those users successfully complete the OLR process.

The following information is recorded by the Authentication Server whenever a new user is added to a SmartGate community:

- User ID
- 128-bit authentication key (entered as 32 hexadecimal characters)
- Group associated with this end user
- User's long name (e.g., John Doe)
- Status flags (whether the account is enabled or disabled)
- Authentication type
- Registration date

The user's record is created with placeholders for additional information recorded at different times. These fields are:

- Last log-in date/time
- Consecutive log-in failure count
- Deletion date

User authentication is performed by either a challenge/response procedure or via validation of a session ticket created by SmartPass on behalf of a user requesting access. The user's 128-bit authentication key is the cornerstone of the authentication process. A duplicate of the 128-bit authentication key, stored on the SmartGate Server, must also be made available to SmartPass during the authentication process. SmartPass' copy of the authentication key may be stored on either a physical smart card or a virtual smart card (soft token).

Information regarding which specific resources each user will be allowed to access via SmartGate is stored in two files—`sgate.ac1` and `sweb.ac1`. These resource descriptions are also referred to as TCP and Web access permissions. The `sweb.ac1` file describes access permissions to Web Server resources, while the `sgate.ac1` file describes access permissions for all non-Web resources. SmartAdmin can be used by the SmartGate administrator, either remotely or locally on a SmartGate Server running on Microsoft Windows NT, to easily manage TCP and Web access permission for both groups and individual users. For more information on managing access permissions using SmartAdmin, see Chapter 5, "Using SmartAdmin."

The following is a detailed description of the layout of these files and exactly how they function. The information recorded in both `sgate.ac1` and `sweb.ac1` is grouped into sections. Each section consists of a section name enclosed in square brackets followed by zero or more access permissions. A file template follows:

```
[section-name1]
access permission1
access permission2
.
.
[section-name2]
access permission1
.
.
```

Sections define access permissions for groups of users or individual users. The section name preceding an individual user's access permissions is the User ID, the same character string used to identify that user's 128-bit authentication key. This name is created either by OLR, a UID Server, or it is assigned by the SmartGate administrator whenever a user

**WARNING!** To create a valid new group, you must use SmartAdmin or the command line; do not edit a file directly.

**NOTE:** There is unlimited nesting of groups.

**NOTE:** Assigning permissions to groups avoids duplication and is more efficient than assigning permissions to individual users.

**NOTE:** Group names can be up to 23 characters, are case-sensitive, and cannot contain spaces or special characters (\*,/,).

account is added manually at the console or via secure remote administration.

The section name used to define access permissions available to more than one user is a group, an arbitrary character string assigned by the SmartGate administrator. The first character of a group must be the “tilde” (~) character. The remaining characters of this name should be a unique string describing the group’s identity or function, such as [~engineering] or [~project123].

Groups make it possible to bundle permissions and assign them together. Groups are typically designed to aggregate services in ways that are most useful to an organization. Aggregating services geographically, for example, works well for an organization that is geographically dispersed. The Chicago and New York LANs are examples of services grouped geographically.

There is one special group, [~all], to which all members of a SmartGate community belong by default.

An `sgate.ac1` or `sweb.ac1` file might contain the following structure:

```
[~all]
access permission1

[jsmith]
~project123

[mjones]
access permission2

[~project123]
access permission3

[~sales]
access permission4

[~engineering]
~project123
access permission5
```

Two other files associated with the authentication process are `sgate.dny` and `sweb.dny`. These files contain lists of access permissions to which access will be “denied” whenever a user’s access permission matches an entry stored in either of these files.

Whenever SmartPass requests authentication and service access on behalf of a user’s application, the deny list in either `sgate.dny` or `sweb.dny` is searched for an access permission

match. If an access permission match is found, access is denied. If no match is found, `sgasrv` searches the access permissions in either `sgate.acl` or `sweb.acl`.

All links to groups within that group are also searched and if an access permission matches, within any level of group, access is allowed.

## Remote Authentication Server

Setting up a remote Authentication Server (`sgasrv`) allows an administrator to configure multiple SmartGate Servers to connect to a single Authentication Server.

To configure the SmartGate Server where the Authentication Server will reside, select the **Remote Authentication Server** check box in SmartAdmin. In the **Authentication client hosts** setting, type a list of SmartGate Servers from which the Authentication Server will allow connections. The Authentication Server only accepts requests from `localhost` and the hosts specified by this setting. The number of hosts you may specify is limited to the maximum line length (255).

To configure a SmartGate Server to connect to a remote Authentication Server, select the **Remote Authentication Server** check box in SmartAdmin. Enter the hostname or IP address of the remote Authentication Server in the **Authentication Server host** text box. This setting is used by the SmartGate proxies when they are not on the same host as the Authentication Server. By default, the Authentication Server listens to port 3838 for requests from the SmartGate Server.

All programs that access the user database (i.e., the Dynamic Configuration Server, UID Server, KRAKit Server, etc.) must reside on the same computer where the user database is stored (i.e., the Authentication Server).

## Server Redirection Access Permissions

Access permissions that specify Server Redirection only apply when used with the Remote Authentication Server.

The SmartGate Server where the Authentication Server resides also maintains the Access Control Lists and the user database. However, an administrator may specify which of the multiple SmartGate Servers should be used by SmartPass to connect to a particular destination. SmartGate provides this capability by allowing an administrator to enter rules into `sgate.acl` and `sweb.acl` that specify which SmartGate Server should build secure paths for a given host name or host mask.

**NOTE:** Refer to the section, **Backup Server Host**, item #2, for the files that must be copied to the remote Authentication Server.

**NOTE:** Using SmartAdmin, select the **Configuration** tab, and then the **Authentication** button.

**NOTE:** Only IP addresses (no hostnames) can be used in the **Authentication client hosts** and the **Server host** settings if **Reverse DNS lookups** is not selected.

**NOTE:** The configuration settings for the a remote Authentication Server are `sgasrv` and `sgasrv_clients`, located in `sgconf.ini` in the SmartGate Server's root directory on a UNIX-based server and SmartGate Server's root directory\data on Windows NT.

**NOTE:** The Server Redirection feature is available in SmartGate 2.7 and SmartPass 3.4 or later.



The following are examples of TCP and Web connections routed to a specified destination through the SmartGate Server `sg-server-to-use`.

`sweb.acl` rule:

```
@sg-server-to-use/www.destination.com/
```

`sgate.acl` rule (a mail connection):

```
@sg-server-to-use/destination.com pop3
```

For example, an administrator wishes to provide secure access to his/her two Internet domains (`domain1.com` and `domain2.com`) using a single user database. The user database is configured on a master Authorization Server, `master-auth.outside.com`. The administrator then configures two SmartGate Servers, one with access to each domain, and configures them to use the master Authorization Server, `master-auth.outside.com`, as their Authorization Server. With this setup, the following rules in `sweb.acl` would route all Web requests for servers behind `domain1` through the SmartGate Server, `sg1.domain1.com`, and all `domain2` requests through the SmartGate Server, `sg2.domain2.com`:

`sweb.acl`:

```
@sg1.domain1.com/*.domain1.com/  
@sg2.domain2.com/*.domain2.com/
```

## Backup Server Host

An additional computer, other than the one running the Authentication Server, can be set up as a backup host storing a redundant user database. If there is a failure of the Authentication Server host, the redundant user database can then take over with minimum time lost. All changes to the user database (`sgusrdb`) will be simultaneously mirrored in the local database on the backup server host.

To set up a redundant user database:

1. Do a complete installation of the SmartGate Server software on a separate computer, excluding the V-ONE license.

The hostname of your backup server should be different from your main SmartGate Server. For instance, if your SmartGate Server is `ambrosia.fence.v-one.com`, your backup host may be `ambrosia2.fence.v-one.com`.

2. You should configure your backup host exactly the same as your SmartGate Server. All the configuration files on your SmartGate Server should be copied to the backup host:
  - All key files, i.e., your public/private key pair and certificate (*keyname.pub*, *keyname.prv*, *keyname.cer*)
  - All access control lists (acls) and deny lists (*sweb.acl*, *sweb.dny*, *sgate.acl*, *sgate.dny*, *adm-gw.acl*, *sgshim.acl*)
  - Configuration options including system definitions, OLR data requirements, and single-port mapping rules (*sgconf.ini*, *reginfo.dat*, and *sgproxy.conf*)
  - The *secret.key* file containing your shared secret key
  - Any UID Server files (Rules File)
3. Using SmartAdmin, connect to your main SmartGate Server. Configure the following settings:
  - Make your backup server a client of your SmartGate Server. Also using SmartAdmin, add the hostname or IP address of your backup server host to the **Authentication client hosts** setting.
  - Type the hostname or IP address and the port number of the backup host in the **Backup server host and port** text box. While the port number is configurable, the redundant database has a default port number of 3901, which is recommended.
4. If the machine running the SmartGate Server fails for any reason, change the host name and IP addresses of the Backup server to that of the original Authentication Server. Install your V-ONE License Key(s) on to your backup host.
5. As soon as the License Key is installed, you should be up and running and the new SmartGate Server will look identical to your end users.

This feature is optional. Unless a backup host has been set up and this option is specified, no backup will be created. If this option is turned on with an existing database, first copy the existing database to the backup host because mirroring of the database occurs in real-time.

**NOTE:** For each server, the *secret.key* file must be copied into the */usr/smartgate/etc* directory on UNIX and the *\Program Files\one\smartgate 4.x\data* directory on Microsoft Windows NT.

**NOTE:** Using SmartAdmin, select the **Configuration** tab, and then the **Authentication** button.

**NOTE:** All configuration settings for the backup host are located in *sgconf.ini* in the SmartGate Server's root directory/*etc* on a UNIX-based server and SmartGate Server's root directory\*data* on Windows NT.

**WARNING!** The SmartGate Server must be rebooted after this option is initiated.

**WARNING!** When using SecurID authentication on a Windows NT SmartGate Server with two network adapter cards, the “default” adapter card cannot be the outside adapter. Ping the WINS Server name (hostname only, without domain name). If the IP address returned is the internal interface IP, then you are all set. If not: Double-click the **Network** properties icon in the **Control Panel**; then click the **Protocol** tab, select **TCP/IP**, and click **Properties**. Reassign the **IP Address**, **Subnet Mask**, and Gateway entries for the adapters that show in the **Adapter** pull-down line.

**NOTE:** The `sdconf.rec` file holds the connectivity information between the SmartGate/RSA SecurID Server and the ACE/Server (i.e., IP address, encryption algorithm, etc.).

**NOTE:** RSA SecurID authentication is available only when using SmartPass 3.1 or later.

## Using RSA SecurID for User Authentication

To use the [RSA SecurID authentication](#) method from RSA Security, Inc. you must do the following:

- Make your SmartGate Server a ‘client’ of your SDI ACE/Server
- Configure your SmartGate Server to run the `sgsdi` service
- Configure the RSA SecurID settings on your SmartGate Server

## Making SmartGate an ACE Client

Making your SmartGate Server a ‘client’ of your RSA ACE/Server requires you to register your SmartGate Server with the ACE/Server (See RSA Security, Inc. documentation on registering servers). Copy the file called `sdconf.rec` from your ACE/Server to the SmartGate Server’s root directory\data on a Windows NT Server or to the SmartGate Server’s root directory/etc on a UNIX-based server.

## Running the `sgsdi` Service

Configuring your SmartGate Server to run the `sgsdi` service is done as follows:

### Microsoft Windows NT:

1. Stop the SmartGate Service (Go to **Control Panel, Services, SmartGate Server**, and click **Stop**).
2. Open a command prompt in the SmartGate Server directory.
3. Enter the command:  
**`sgsdi -regserver`**
4. Start the SmartGate Service (Go to **Control Panel, Services, SmartGate Server**, and click **Start**).
5. To remove services, open a command prompt in the SmartGate Server’s root directory and enter the command:

**`sgsdi -unregserver`**

### UNIX-based:

1. Use the setup script from the SmartGate Server software by running `./setup` in the /SmartGate Server’s root directory/bin.

2. On the SmartGate Server Software Main Menu, press **C** for configuration options.
3. Press **E** in the SmartGate Server Setup Menu to open the SmartGate Extensible Components Menu.
4. To enable the SecurID services, press **D**. RSA SecurID is disabled by default.

### Additional UNIX Command Line Options

In order to run the SmartGate RSA SecurID Service in the foreground on the command line, use the following format :

```
/usr/smartgate/libexec/sdiServer -flag
```

Some of the most commonly used flags include:

- p [port]** runs the RSA SecurID daemon on TCP port [port] (default 2095)
- v** displays RSA SecurID version information
- h** displays all RSA SecurID options

In order to put the Service in the background as a daemon use the following:

```
/usr/smartgate/libexec/sdiServer -p 2095 &
```

The **-p** port flag is optional. If the port is not specified, then SmartGate RSA SecurID Service will look in the `/etc/services` file to assign the port.

In order to make the change permanent for SmartGate, you must change the start-up file for SmartGate. The start-up file is located at different location dependent on your operating system and firewall installation.

Solaris:

```
/etc/rc2.d/S90sgate
```

BSD/OS:

```
/etc/rc.local
```

RedHat Linux:

```
/etc/rc.d/rc3.d/S90sgate
```

The changes you make to the start-up file for SmartGate WILL NOT be saved during an upgrade of the SmartGate Server software.

**NOTE:** If you change the default RSA SecurID TCP port (2095), you must make the corresponding change for the SmartPass RSA SecurID client. Do this via SmartAdmin.

**WARNING!** When using RSA SDI authentication with two network adapter cards, the “default” adapter card cannot be the outside adapter. Using the Windows NT Network Adapter setup, switch the adapter order for the IPs used.

**NOTE:** If you want your end users to be automatically enabled after registration you must configure the ACE/Server. The SmartGate Server setting, **New OLR users enabled**, does not function with RSA SecurID authentication.

**NOTE:** RADIUS is an open-standard (RFC 2138) authentication protocol and is transported over UDP only, not TCP.

**WARNING!** There must be a clear path of communication for bidirectional UDP traffic from your SmartGate/RADIUS Server to the RADIUS Backend Server.

## Configuring the SmartGate Server

There are two settings in `sgconf.ini`, located in the SmartGate Server’s root directory, that affect RSA SecurID authentication. These entries can be configured using SmartAdmin (see Chapter 5, “Using SmartAdmin”). Open SmartAdmin, click the **Configuration** tab, and then click **Other**. The following options will be listed in alphabetical order:

### `sdi_timeout`

This setting specifies the number of minutes that the end user, when using RSA SecurID authentication, will be allowed before responding to a **Next Tokencode** or **New Pin Code** dialog box. The valid range is 1 to 30 minutes and the default is 3 minutes.

### `sdi_ttl`

This setting specifies the number of minutes for which a RSA SecurID authentication will remain valid before another authentication is required. The valid range is 1 to 1440 minutes and the default is 30 minutes. Do not use commas.

## Using RADIUS for User Authentication

To use the **RADIUS authentication** method you must do the following:

- Configure your SmartGate Server to run the `sgradius` service
- Configure the RADIUS settings on your SmartGate Server
- Configure the RADIUS Backend Server to recognize your SmartGate Server running the RADIUS module

## Running the `sgradius` Service

Configuring your SmartGate Server to run the `sgradius` service is done as follows:

### Microsoft Windows NT:

1. Stop the SmartGate Service (Go to **Control Panel, Services, SmartGate Server**, and click **Stop**).
2. Open a command prompt in the SmartGate Server directory.
3. Enter the command:

**`sgradius -regserver`**

4. Start the SmartGate Service (Go to **Control Panel, Services, SmartGate Server**, and click **Start**).
5. To remove services, open a command prompt in the SmartGate Server root directory and enter the command:

**sgradius -unregserver**

#### **UNIX-based:**

1. Use the setup script from the SmartGate Server software by running `./setup` in the /SmartGate Server's root directory/bin.
2. On the SmartGate Server Software Main Menu, press **C** for configuration options.
3. Press **E** in the SmartGate Server Setup Menu to open the SmartGate Extensible Components Menu.
4. To enable RADIUS authentication services, press **R**. RADIUS is disabled by default.

#### **Additional UNIX Command Line Options**

To run the SmartGate RADIUS Service in the foreground on the command line, use the following format:

```
/usr/smartgate/libexec/radiusServer -flag
```

Some of the most commonly used flags include:

- p [port]** runs the RADIUS daemon on TCP port [port] (default 2097)
- v** displays RADIUS version information
- h** displays all RADIUS options

In order to put the service in the background as a daemon use the following:

```
/usr/smartgate/libexec/radiusServer -p 2097 &
```

The `-p` port flag is optional. If the port is not specified, then SmartGate RADIUS Service will look in the `/etc/services` file to assign the port.

In order to make the change permanent for SmartGate, you must change the start-up file for SmartGate. The start-up file is located at different location dependent on your operating system and firewall installation.

Solaris:

```
/etc/rc2.d/S90sgate
```

**NOTE:** If you change the default RADIUS TCP port (2097), you must make the corresponding change for the SmartPass RADIUS client. Do this via SmartAdmin.

**NOTE:** To receive complete RADIUS debugging information from the SmartGate Server into the SmartGate log, the `sgconf.ini` setting, **Debug reporting**, must be level '4' or above. The default for this setting is '0.'

**NOTE:** If you want your end users to be automatically enabled after registration you must configure the RADIUS Backend Server. The SmartGate Server setting, **New OLR users enabled**, does not function with RADIUS authentication.

BSD/OS:

`/etc/rc.local`

RedHat Linux:

`/etc/rc.d/rc3.d/S90sgate`

The changes you make to the start-up file for SmartGate WILL NOT be saved during an upgrade of the SmartGate Server software.

## Configuring the SmartGate Server

There are additional settings in `sgconf.ini`, located in the SmartGate Server's root directory, that affect RADIUS authentication. These entries can be configured using SmartAdmin (See Chapter 5, "Using SmartAdmin"). Select the **Configuration** tab and click **RADIUS**. The following options are displayed:

### RADIUS Backend Servers: Host (`radius_authsrv[1...5]`)

The FQDN or IP address of the RADIUS Backend Server and backup RADIUS Servers (up to 5). If RADIUS is being used, at least one server must be specified as the RADIUS Backend Server. Additional backup servers are optional. There is no default. An example from the `sgconf.ini` file follows:

**Example:** `radius_authsrv1=10.0.0.225`  
`radius_authsrv2=10.0.0.226`

### RADIUS Backend Servers: Secret (`radius_authsrv[1...5]_secret`)

The shared secrets for the RADIUS Backend Server and each of its backups (up to 5). If RADIUS is being used, at least one value must be specified for the RADIUS Backend Server. Additional backup servers are optional. There is no default. An example from the `sgconf.ini` file follows:

**Example:** `radius_authsrv1_secret=1tzieojh54343`  
`radius_authsrv2_secret=ol34kduf67`

Each RADIUS Backend Server must be configured with its corresponding shared secret code. See your RADIUS documentation for further information.

## RADIUS Backend Servers: Use CHAP

### (radius\_authsrv[1...5]\_usechap)

If your RADIUS Backend Server is using CHAP authentication for its users, the SmartGate/RADIUS Server will simulate a CHAP exchange and send a CHAP-Password value instead of the normally hashed User-Password attribute. Each of the servers are defaulted to “no.” An example from the `sgconf.ini` file follows:

**Example:** `radius_authsrv1_usechap=yes`  
`radius_authsrv2_usechap=no`

## RADIUS Backend Servers: Wait

### (radius\_authsrv[1...5]\_waitfor)

The wait time (in seconds) for requests made to the RADIUS Backend Server before it times out. Network factors may prevent certain servers from responding as quickly as they should. The default is 120 seconds for each server with a maximum of 32767 seconds. An example from the `sgconf.ini` file follows:

**Example:** `radius_authsrv1_waitfor=120`  
`radius_authsrv2_waitfor=23859`

## Time to live (radius\_ttl)

The number of minutes that a RADIUS authentication will remain valid before another authentication is required. The valid range is 1 to 1440 minutes and the default is 30 minutes. An example from the `sgconf.ini` file follows:

**Example:** `radius_ttl=60`

## Challenge timeout (radius\_challenge\_timeout)

The number of minutes that a RADIUS challenge dialog box will remain on the screen before it times out. The valid range is 1 to 30 minutes and the default is 5 minutes. An example from the `sgconf.ini` file follows:

**Example:** `radius_challenge_timeout=10`

## Configuring the RADIUS Port

Some older RADIUS Servers, such as Livingstone RADIUS, were published before a revision of the RADIUS protocol port. As documented in RFC 2138 (<http://www.ietf.org/rfc/rfc2138.txt>), the initial incorrect port chosen for RADIUS is 1645, and the correct RADIUS port is 1812. These older servers, therefore, operate on UDP port 1645, which is outdated.

**NOTE:** Do not use commas in the **RADIUS Authentication Servers: Wait** setting.



SmartGate chooses which port to proxy RADIUS on by first checking the system services table for an entry for RADIUS. If no entry is found, SmartGate defaults to UDP port 1812. On Windows NT, the system services table is located in `C:\WINNT\system32\drivers\etc\services`. If you have one of these older RADIUS Servers, add the following service table entry to force SmartGate to proxy the RADIUS protocol on port 1645 rather than port 1812:

```
radius    1645/udp    # Legacy RADIUS protocol
                        (incorrect port, see RFC2138)
```

## Configuring the RADIUS Backend Server

The shared secret code for each RADIUS Backend Server must match the codes assigned on the SmartGate/RADIUS Server.

The shared secret code(s) are set using the `radius_authsrvn_secret` option in `sgconf.ini`.

Refer to your RADIUS documentation for additional information.

## SmartPass/RADIUS Backend Server Interaction

SmartPass submits a request to the SmartGate Server running the RADIUS module. It, in turn, forwards the request to the RADIUS Backend Server.

The arguments are:

- **USERID**                RADIUS User ID
- **PASSCODE**            RADIUS user passcode
- **KEY**                    Randomly-created shared secret key to be sent to the SmartGate Server for the session to be activated

Responses are:

- **SUCCESS**            RADIUS authentication was successful. Contains TTL field retrieved from the server to determine how long current SmartGate/RADIUS session lasts
- **ETIMEOUT**            Response from the server timed out
- **ENETINIT**            Network initialization error

- **EINTERNAL**      Some sort of failure occurred internally on the SmartGate/RADIUS Server, such as a memory allocation problem
- **EMALFORMED**    Response from the server was malformed or unrecognizable
- **EAUTH**            User failed authentication

Each error token will have a string value, **REASON**, that contains a message from the SmartGate/RADIUS Server. It will provide a complete error response that can be printed out and displayed on SmartPass. It will also contain a **CODE** field that provides an error number for reference.

- **CHALLENGE**      The SmartGate/RADIUS Server received an Access-Challenge request from the RADIUS Backend Server. This response must contain two tokens: a text field, **SERVERRESPONSE**, which contains the RADIUS Backend Server's response, and **TIMEOUT**, which contains the maximum time (in minutes) that the client can wait for the challenge to the server.

## Using Entrust for User Authentication

To use the [Entrust authentication](#) method you must do the following:

1. Install and configure your Entrust software.
2. Check the `entrust.ini` file and configure if necessary.
3. Install the SmartGate Server software.  
Configure the SmartGate Server software with the location ([directory](#)) of the `entrust.ini` file, [Entrust's Reference number](#), and [Entrust's Authorization code](#). If needed, start the Entrust/Netrust Service on the SmartGate/Entrust Server (Windows NT only).
4. Configure the SmartGate Server.
5. Prepare the SmartPass Installation package for distribution to the end users.

**WARNING!** The system time on the [Entrust CA Server](#), the [SmartGate/Entrust Server](#), and the computer where SmartPass is running must be set within 5 minutes of each other.

**NOTE:** Your Entrust CA Server may run on the same computer where the SmartGate Server will reside.

**NOTE:** You will need the location of `entrust.ini` and the run time libraries during installation of the SmartGate Server software.

## Installing and Configuring the Entrust Software

In order to use Entrust as your authentication method, the Entrust CA Server software must first be installed and configured and your SmartGate Server must be added as a user. See your Entrust documentation for complete instructions on adding a user. From the Entrust CA Server, make note of the SmartGate Server's:

- Reference number
- Authorization code

You will need them during installation of the SmartGate Server software.

The following files must be present on the machine where the SmartGate Server will reside before installation of the SmartGate Server software:

- `entrust.ini`
- Run time libraries:

### Windows NT

<code>etmem32.dll</code>	<code>etsesn32.dll</code>
<code>msvcirt.dll</code>	<code>msvcrt.dll</code>

### UNIX—Solaris operating system

<code>libEntrustSW.so</code>	<code>libetsesnSW.so</code>
------------------------------	-----------------------------

If the “EntrustSession Toolkit” has been installed or if the Entrust CA Server is installed on the same machine, all necessary files will be present.

## Configuring the `entrust.ini` File

The `entrust.ini` file is included with your Entrust software. This file contains the location of the Entrust CA Server and Manager and is used by both the SmartGate Server and SmartPass. Check this file for proper configuration by performing the following steps:

1. Open the `entrust.ini` file using a text editor such as Notepad.
2. Make sure the Manager and Server settings are configured with the IP address and port number of your Entrust CA Server.
3. Check the `DefaultProfileLocation` setting.

This setting, when used, specifies which directory the `.epf` file will be written to during generation and where the SmartGate/Entrust Server will look for it during the authentication process.

4. If this line has already been set to a specific directory, you should leave it alone, especially if there are multiple services using Entrust.
5. If this line has been commented out or the value is not set; it should be uncommented and the location should be set to:

**Windows NT** - C:\SmartGate Server's root directory\

**UNIX** - /SmartGate Server's root directory/etc/

The following is an example of the first few lines of an `entrust.ini` file on a Windows NT Server which has been set to the default SmartGate directory:

```
[Entrust Settings]
Manager=10.0.0.138+709
Server=10.0.0.138+389
DefaultProfileLocation=C:\Program Files\V-ONE\
SmartGate 4.0\
```

The machine where your SmartGate/Entrust Server will reside and the SmartPass installation disk #1 must both have a copy of `entrust.ini`.

## Installing the SmartGate Server Software

Depending on the operating system on which the SmartGate Server software is running, installation and activation of Entrust authentication differs slightly.

### Microsoft Windows NT SmartGate Server

During installation of the SmartGate Server software on Microsoft Windows NT, you will be asked if you want to install the Entrust/Netrust components. Click **yes**. Three windows will be presented, in which you will:

1. Enter or browse for the `entrust.ini` file. The `entrust.ini` file is obtained from Entrust—not V-ONE. It is necessary for the operation of Entrust authentication.
2. Type the Reference number issued by the Entrust CA Server.
3. Type the Authorization code issued by the Entrust CA Server.

When you choose to install Entrust during a Windows NT installation, the Entrust/Netrust Service is automatically started.

**NOTE:** The ports displayed are the Entrust CA Server's default ports.

**NOTE:** For information on removing services, see [“Adding and Removing Services”](#) in Chapter 4, “Installing the SmartGate Server Software - Windows NT.”

## Activating Entrust At a Later Time

If, during installation of the SmartGate Server software, you selected **no** when asked if you wanted to install Entrust, you may activate Entrust at a later time by manually starting and registering the Entrust/Netrust Service. You will also need to generate an `.epf` file using the location of your `entrust.ini` file combined with your Entrust Reference number and Authorization code.

The following steps need to be performed:

1. Start the Entrust/Netrust Service:

Open a command prompt in the SmartGate Server root directory and type the command:

**`sgent -regserver`**

Type: **`ntrsvc -install`**

2. Generate an `.epf` file:

Type **entCreate**, the full path name of the `entrust.ini` directory enclosed within double quotes, the Reference number, and the Authorization code—with one space separating each item. For example:

```
entCreate "c:\Program Files\V-ONE\SmartGate  
4.0\entrust.ini" <space>01106044<space>VX8V-L4TA-SALM
```

3. Start the Entrust/Netrust Service (Go to **Control Panel, Services, ENTRUST Service**, and click **Start**)—or Reboot.
4. Restart the SmartGate Service (Go to **Control Panel, Services, SmartGate Service**, click **Stop** and then **Start**).
5. To remove the Entrust/Netrust Service:

Stop the ENTRUST Service (Go to **Control Panel, Services, ENTRUST Service**, and click **Stop**).

Open a command prompt in the SmartGate Server root directory and enter the command:

**`sgent -unregserver`**

Type: **`ntrsvc.exe -remove`**

## UNIX-Based SmartGate Server

If you are installing the SmartGate Server software on Solaris, you will need to activate and configure the Entrust module from the setup script. Use the following steps:

1. Run the setup script from the SmartGate Server software by running `./setup` in the SmartGate Server's root directory/bin.
2. On the SmartGate Server Software Main Menu, press **C** for configuration options.
3. Press **E** in the SmartGate Server Setup Menu to open the SmartGate Extensible Components Menu.
4. To enable Entrust, press **E**. Entrust is disabled by default.
5. Press **N** to enter the Entrust/Netrust Configuration Menu.
  - Press **L** and enter the directory where the run time libraries are located. These files are obtained from Entrust—not V-ONE. The run time libraries are necessary for the operation of Entrust.
  - Press **I** and enter the path for the `entrust.ini` file.
  - Press **R** to register your SmartGate Server with the Entrust/Netrust Authority. You will be prompted to enter the Entrust Reference number and Authorization code obtained from your Entrust CA Server.
  - Press **E** to specify either the hostname or IP address and the port number of the optional SmartGate UID Server for an Entrust UID Server. This option configures the `uid_server[Entrust]` setting in `sgconf.ini`.

### Additional UNIX Command Line Options:

SmartGate Entrust/Netrust Service requires the location of the `entrust` session libraries and the `entrust.ini` file defined in environment variables. To run any of the SmartGate Entrust/Netrust services, these variables must be set. The example below is for Bourne Shell users:

```
export LD_LIBRARY=/usr/local/entrust/lib
export ENTRUST_INIFILE=/usr/smartgate/etc/entrust.ini
```

In order to run the SmartGate Entrust/Netrust Service in the foreground on the command line, use the following format:

```
/usr/smartgate/libexec/entServer -flag
```

**NOTE:** For more information, see “[Entrust/Netrust Authentication](#)” in Chapter 3, “Installing the SmartGate Server Software - UNIX.”

**NOTE:** Using SmartAdmin, select the **Configuration** tab, and then the **On-Line Registration** button.

Some of the most commonly used flags include:

- p [port]** runs the Entrust daemon on TCP port [port] (default 2096)
- v** displays Entrust version information
- h** displays all Entrust options

In order to put the service in the background as a daemon use the following:

```
/usr/smartgate/libexec/entServer -p 2096 &
```

The **-p** port flag is optional. If the port is not specified, then SmartGate Entrust/Netrust Service will look in the `/etc/services` file to assign the port.

In order to make the change permanent for SmartGate, you must change the start-up file for SmartGate. The start-up file is located at different location dependent on your operating system and firewall installation.

Gauntlet Firewall for Solaris:

```
/usr/local/etc/mgmt/rc/S400sglic
```

Solaris (other than Gauntlet Firewall):

```
/etc/rc2.d/S90sgate
```

The changes you make to the start-up file for SmartGate WILL NOT be saved during an upgrade of the SmartGate Server software.

## Configuring the SmartGate Server

Before your end users can perform OLR using Entrust as their authentication method, the SmartGate Server must be configured. Perform the following steps:

1. Allow for Entrust as a registration method by selecting the **ENTRUST** check box in addition to the **V-ONE** check box, in the **OLR methods** option.
2. Set up a separate set of Entrust registration fields by customizing the OLR registration file (`reginfo.dat`). These fields will be requested of the Entrust end user during OLR. Open SmartAdmin, click the **OLR Setup** tab. The default V-ONE registration fields are displayed. Click the **Add** button, and select **ENTRUST** in the **Field Page** section to enter Entrust-specific registration fields. You may enter up to 10 Entrust data entry fields.

You may set up a UID Server if you want to assign specific User IDs to individual users or use an existing database of users. As administrator, you also have the option of creating your own UID Server process. For detailed instructions on setting up a UID Server, see “[UID Server for On-Line Registration](#)” in Chapter 7, “On-Line Registration Services.”

## Preparing the SmartPass Installation Package for Entrust Authentication

To configure the SmartPass installation package for Entrust authentication, the following steps must be performed:

1. Configure the `Packages` option for Entrust authentication in the `setup.ini` file located on the SmartPass installation disk(s).
2. Copy the `entrust.ini` file onto disk 1 of the SmartPass installation disk(s)—Optional.
3. Create the `entrust.z` archive with files obtained from your licensed Entrust software and copy it onto the SmartPass installation disk(s)—Optional.
4. Deploy the SmartPass software on disk or on a company Web site as a downloadable file.
5. Distribute to your end users either an existing `.epf` file or an Entrust Authorization code and Reference number.

### Configuring `setup.ini`

The SmartPass installation package can be adjusted by the presence of specific authentication options and programs. Depending on what type(s) of authentication your end users will be using, you may wish to remove support for unused tokens. This will reduce the size of the distribution package and simplify installation. A standard installation is defaulted to automatically include the FIPSTKN, MCOS, VCAT, and SHIM options.

In order to include Entrust authentication on the SmartPass installation disk, `SGENTRUS` and `ENTRUST` must be specified in the `Packages` option in `setup.ini`. If you want your end user to use Entrust authentication exclusively, configure the `Packages` option for **ENTRUST** and **SGENTRUS** only. For example:

```
Packages=SGENTRUS, ENTRUST
```

The `SGENTRUS` option signifies the Entrust component of the SmartGate Server software (`sgentrus.z`). The `sgentrus.z`

**NOTE:** The `entrust.ini` file and the `entrust.z` archive are necessary for SmartPass to run using Entrust authentication. Only under the circumstances outlined in the following sections, “[Copying the entrust.ini File](#)” and “[Preparing the entrust.z Archive](#)” would they be omitted.

**WARNING!** In order for your SmartPass end users to create an `.epf` file, they must have their own Entrust Authorization code and Reference number issued by the Entrust CA Server.

**WARNING!** The Entrust authentication options are not included in the default SmartPass installation package. They must be specified by the SmartGate administrator in the `setup.ini` file.



archive is supplied by V-ONE and included on the SmartPass installation disk(s) contained on your SmartGate CD-ROM.

However, the ENTRUST option signifies the `entrust.z` archive, which is created by the SmartGate administrator and copied onto the SmartPass installation disk. See “Preparing the `entrust.z` Archive” for instructions.

### Copying the `entrust.ini` File

Copy the `entrust.ini` file from the SmartGate/Entrust Server onto the SmartPass installation disk #1. You should have already checked and possibly configured this `entrust.ini` file before installing the SmartGate Server software, as described in “Configuring the `entrust.ini` File” earlier in this chapter.

During installation of the SmartPass software, the `entrust.ini` file will be written to the SmartPass directory. If the end user is upgrading, the new `entrust.ini` file will overwrite the old one. If you want them to use their existing SmartPass `entrust.ini`, do not include it on the installation disk.

### Preparing the `entrust.z` Archive

If ALL your end users have the EntrustSession toolkit installed on the machines where they will be running SmartPass, you do not need to include the `entrust.z` archive on the SmartPass installation disk. The EntrustSession toolkit contains all necessary files.

Otherwise, you must create this archive with files obtained from your licensed Entrust software; and then copy it to the first SmartPass installation disk.

#### Tools Needed:

You will need the program ICOMP.EXE from InstallShield 3.0 or InstallShield Express in order to prepare this compressed archive.

The contents of the `entrust.z` archive are NOT supplied by V-ONE:

<code>etmem32.dll</code>	<code>etsesn32.dll</code>
<code>msvcirt.dll</code>	<code>msvcrt.dll</code>

The archive must also include a file called `entrust.inf` which is used by the installation script. This file should be an ASCII file containing only a semi-colon (;) since no special installation instructions are required.

## Instructions:

Use the following instructions to prepare the `entrust.z` package:

1. `icmp -h etmem32.dll entrust.z`
2. `icmp -h etsesn32.dll entrust.z`
3. `icmp -h msvcirt.dll entrust.z`
4. `icmp -h msvcrt.dll entrust.z`
5. copy `entrust.z` onto the disk image

## Viewing a Distinguished Name

Each Entrust user has a unique “Distinguished Name” which identifies each individual user to the Entrust CA Server.

Every time an end user is authenticated, using the Entrust authentication method, a Distinguished Name file is created. A new Distinguished Name file will overwrite an existing file from a previous authentication. Therefore, all Entrust end users who have been authenticated at least once will have a single corresponding Distinguished Name file on the SmartGate/Entrust Server. These files are saved in:

**Windows NT** - `C:\SmartGate Server's root directory\data\dname`

**UNIX** - `/SmartGate Server's root directory/etc/dname`

The Distinguished Name file appears in the following format:

`dname.userid`

where: *userid* is the end user's User ID

To display an end user's Distinguished Name using SmartAdmin, click the **Users** tab, select the desired user's record, and click **Edit**. The user's Distinguished Name will be displayed as the first line in the **On-Line Registration data** field of the **Edit User** Window regardless of whether the user performs OLR or registers anonymously. The Distinguished Name and the OLR data displayed are for your information only; they cannot be edited.

Unlike the end user's Distinguished Name, the SmartGate/Entrust Server's Distinguished Name is written to the bottom of the `sgconf.ini` file. The name is refreshed every time the server authenticates to the Entrust CA Server.

**NOTE:** PKI authentication requires License code 54.

## Using PKI Authentication for User Authentication

Public-key infrastructure (PKI) is the combination of software, encryption technologies, and services that enable businesses to protect the security of their communications and business transactions via the Internet.

PKI integrates digital certificates, public-key cryptography, and certificate authorities into enterprise-wide network security architecture. SmartGate/PKI encompasses the verification of digital certificates to individual users and servers, OLR for end-user enrollment, and SmartAdmin for managing certificates.

### Benefits of using PKI include:

- Securing global interaction between employees and their corporate Intranet
- Allowing the user to create secure Extranets and Virtual Private Networks that give select partners easy access to business-critical information stored on internal networks
- Facilitating the use of secure *e*-business over the Internet

## Digital Certificates

A **digital certificate** is an electronic “credit card” that establishes user credentials when doing business or other transactions on the Web. A digital certificate is issued by a certification authority (CA), and contains information such as: your name, a serial number, a copy of the certificate holder’s public key (used for encrypting and decrypting messages and digital signatures), expiration dates, and the digital signature of the certificate-issuing authority so a recipient can verify if the certificate is authentic. Most digital certificates conform to a standard, X.509.

Digital certificates do two things:

1. They authenticate the identity of the holders.
2. They protect data and information exchanged on-line from theft or tampering.

## Personal Certificates

Personal certificates allow a user to authenticate a visitor's identity and restrict access to specified content. Personal certificates are also used to send secure e-mail for private account information.

## Adding CA Certificates to the Trusted CA List

Installation of a PKI Server certificate onto the SmartGate Server must be done **via the command line**.

1. Move the CA certificate file into the SmartGate Server's root directory\data on a Windows NT Server or to the SmartGate Server's root directory/etc on a UNIX-based server.

**Microsoft Windows NT:**

**C:\Program Files\V-One\SmartGate 4.1\data\Pki\Ca**

**UNIX-based:**

**/usr/smartgate/etc/pki/Ca**

2. Change directory to:

**Microsoft Windows NT:**

**C:\Program Files\V-one\SmartGate 4.1**

**UNIX-based:**

**cd /usr/smartgate/etc/libexec**

3. Add the CA certificate by running the command, **certmanager.exe**. Be sure to include the complete path of the location of the certificate file.

For example:

**C:\Program Files\V-One\SmartGate 4.1\certmanager.exe  
<space> -a .\data\pki\ca <cert filename>**

## Command Line Configuration Variables for certmanager.exe

certmanager.exe accepts the following command line variables:

-l	View trusted CA list
-a <cert filename>	Add a certificate to trusted CA list
-qd	Query delete all certificates from trusted CA list

# Chapter 7

## On-Line Registration Services

**NOTE:** The UID Server is unavailable for PKI authentication.

SmartGate provides On-Line Registration (OLR) services which you may wish to implement depending on your system configuration and the functional requirements of your organization. Included in this chapter are the following:

- SmartGate Server Configuration for On-Line Registration
- User ID (UID) Server for On-Line Registration
- Manual Setup of an HTML Page for On-Line Registration
- SmartPass Deployability
- Performing OLR through a firewall
- Performing OLR utilizing SmartPass/PKI Token

### SmartGate Server Configuration for On-Line Registration

OLR is configured in SmartGate using two configuration files: `reginfo.dat` and `sgconf.ini`. Information from these two files is used to create the Web form, which SmartPass uses to perform registration.

The `sgconf.ini` file initializes the OLR Client with information that will be used as the boot information for the key to be added to the token during OLR. A key on a smart card is part of the boot information, which is in three parts:

- The address of the SmartGate Server
- An authenticator, which must be unique
- A key, which is dynamically created during OLR

The SmartGate Server name is indicated in the `sgconf.ini` file by the `domainname` tag. These tags are not case sensitive, meaning that the boot information for the key obtained during

OLR contains the value associated with `domainname` as its server name component. For example:

```
domainname=206.205.74.231
```

This setting specifies that `206.205.74.231` should be used as the server name in the boot information during OLR.

For multi-interface SmartGate Servers, a different interface may be specified for use by inside users—for example:

```
InsideIP=10.10.0.231
```

This specifies that inside users will use this address as their server name. The Web page may be configured to use the `InsideIP` tag (or any other tag you want) for the server name instead of `domainname`.

The Authenticator name is entirely up to the administrator of the SmartGate Server but must be unique for all authenticators on a single card. It is specified in the `sgconf.ini` file as:

```
authenticator=authenticator
```

This authenticator is stored on the smart card during OLR.

## UID Server for On-Line Registration

This feature allows you to specify the User ID that SmartGate will generate for each user when the user performs OLR. It is useful for implementing single logon applications with SmartGate Aware Servers where the User ID that SmartGate sends to the application server is the one recognized by the application. In addition to the standard V-ONE UID Server, separate UID Servers can be created for use with Netrust or Entrust authentication. You can also create your own UID Server process and configure it to deny OLR to specific users.

### How This Feature Works

The UID Server for OLR is implemented by the addition of an optional program called `sguidsrv`. The `sguidsrv` program accepts a set of `name=value` fields entered by the user during OLR and returns a specific User ID. That User ID was configured in the Rules File as the ID to be selected for this user depending on the settings of one or more of the `name=value` fields. Although you can configure the On-Line Registration to request that the user enter up to 10 fields for each OLR method during OLR, not all of these fields must be considered when selecting the correct User ID for the user.

**NOTE:** Upon successful OLR the `domainname`, User ID, authenticator, and keyname are returned to the end user.

**NOTE:** If you use the UID Server, you must use the UID Server exclusively; you cannot combine both the use of the UID Server and automatic User ID generation in OLR sessions.

**NOTE:** Please refer to the *SmartGate With Netrust Authentication Guide* for information on Netrust.

**NOTE:** Normally, the OLR Server will generate a User ID that is guaranteed to be unique; with the UID Server, however, you are responsible for guaranteeing uniqueness.

Here is a basic example of how this feature works:

1. You decide that your users must enter data for a field called Username.
2. The user enters his/her first and last names in the field.
3. This value is returned to the OLR Server as Username=Joe Smith.
4. The OLR Server passes Username=Joe Smith to the UID Server.
5. The UID Server searches its Rules File for an entry containing Username=Joe Smith.

If the UID Server finds a match, it returns the User ID value (e.g., j . smith) associated with Username=Joe Smith, followed by a new line, and completes step 6.

If the UID Server does not find a match, it returns a new line only. The OLR Server interprets this as a failure and notifies the OLR Client accordingly, preventing the user from performing OLR.

6. Instead of generating a random User ID as it normally would, the OLR Server uses j . smith as the User ID of that user in all subsequent processing.

## Setting Up Your SmartGate UID Server

To set up your UID Server, perform the following actions:

- Configure your SmartGate Server to use a UID Server
- Create a Rules File
- Register your UID Server

## Configuring Your SmartGate Server

Use of the UID Server is optional. There are two configuration settings that must be set for the UID Server to be active. Open SmartAdmin, click the **Configuration** tab, and then click **On-Line Registration**. The following options will be displayed:

## User ID Servers host and port (uid\_server)

Specifies either the hostname or IP address and the port number of the optional SmartGate UID Servers (sguidsrv). Up to four separate UID Servers can be used depending on the authentication method. This setting is used in conjunction with the **User ID Rules File** setting.

### Format: *host:port*

where: *host* is the hostname or IP address of the machine on which the UID Server resides.

*port* is the port number on which the UID Server listens and by default 3846. If you must use a different port, refer to the following section “Changing Your UID Server Port.”

If the UID Server resides on the same machine as your SmartGate Server, set *host* to `localhost` (`127.0.0.1`). There is no default; if a UID Server is not specified, the feature will not be used.

## Changing Your UID Server Port

### Microsoft Windows NT:

- Click the **Start** button and select **Run**. Type **regedit** and click **OK**. The Windows Registry Editor will be displayed. Select:  
`\\HKEY_LOCAL_MACHINE\SOFTWARE\V-ONE\SmartGate\4.x\extensions\sguidsrv\port`  
Double-click **port**, enter the new port value, and click **OK**.

### UNIX-based:

- Using a UNIX editor, such as `vi` or `pico`, open the `services` file located in the `/etc` directory. Replace `3846` (the default) in the following lines with the new port number:  
`sguidsrv 3846/tcp`

## User ID Rules File (UidFile)

Specifies the location (full path name) of the Rules Files. This setting is used in conjunction with the **User ID Server host and port** setting.

### Format: *location*

where: *location* is the name and full path of the Rules File. Make certain the name of your Rules File matches the name in this setting.

**WARNING!** All programs that use the user database (i.e., the Dynamic Configuration Server, User ID Server, etc.) must reside on the computer where the user database is stored (i.e., the Authentication Server).

**NOTE:** The registry keys can be displayed in either decimal or HEX format.



## Creating a Rules File

The Rules File is a simple text file consisting of a set of records, one per line. You are responsible for creating the Rules File.

Follow these Rules File syntax guidelines:

- Each record contains two components: a match component and a User ID component. Use a semicolon to delimit the two.
- A typical record in the Rules File might look like this:  
`First Name=Joe&Last Name=Smith;j.smith`  
A more complex Rules File record might be:  
`First Name=Joe&Last Name=Smith&phone=555-1212  
&SSN=111223333;j.smith`
- The match component consists of one or more *name=value* pairs. Use an ampersand (&) to connect the pairs.
- All tests must be on one line and a line cannot exceed 511 characters. The number of *name=value* pairs should be only as many as are necessary to guarantee a correct match.
- *name* and *value* may both contain embedded spaces.
- *name* and *value* may not contain either leading or trailing spaces.
- You may use spaces before the User ID (after the semicolon) to improve readability, but the User ID itself may not contain embedded spaces.
- The Rules File records do not have to be in any special order.

## Guidelines for Matching

- During OLR, the UID Server will read the records in the Rules File sequentially from the beginning until a match is found between the *name=value* pairs in the file and those received from your user via the OLR Server.
- All matching is case-blind.
- All of the *name=value* pairs in the Rules File record must be matched by *name=value* pairs entered by your user. However, not all of the *name=value* pairs entered by your user have to be matched. For example, you may decide that the matching will be performed solely in terms of Social Security Number. In this case, a record in the Rules File might contain:

```
ssn=111223333; user.id
```

**NOTE:** Use "group=" for a group list.

and the data received from the user might consist of other information, such as:

```
First Name=Joe&Surname=Smith&phone=555-1212&
SSN=111223333;j.smith
```

The unused *name=value* pairs (in this case, First Name, Last Name, and phone) are ignored.

- Wildcards can be used and if the User ID is wild, then the last wildcard string will be the User ID.

```
FirstName=joe&Surname=Smith&PinNumber=*;joe
FirstName=joe&PinNumber=*;*
```

## Registering Your UID Server

### Microsoft Windows NT

The UID Server (sguidsrv) is running by default—no further setup is required. See “[Adding and Removing Services](#)” in Chapter 4, “Installing the SmartGate Server Software - Windows NT,” for more information.

### UNIX-based

The UID Server is normally run as a daemon invoked by `inetd`. To allow this:

1. Edit the `/etc/inetd.conf` file and uncomment the `sguidsrv` line:

```
sguidsrv stream tcp nowait root
/usr/smartgate/libexec/sguidsrv sguidsrv
```

2. Restart `inetd` by rebooting or signal `inetd` by typing:

Solaris:

```
ps -ef|grep inetd
kill -HUP process_id
```

BSD/OS:

```
ps -ax|grep inetd
kill -HUP process_id
```

where: the *process\_id* number for `inetd` is returned by the system after typing the first command as described above.

**NOTE:** Using SmartAdmin, select the **Configuration** tab, and then the **On-Line Registration** button.

**NOTE:** The UID Server can run on any machine accessible via TCP and you can use any UID Server that uses that protocol.

## Creating Your Own UID Server Process

If you want to write your own UID Server process to generate User ID's, you must:

1. Configure the **User ID Servers host and port** (`uid_server`) setting so that it points to your process.
2. Create a TCP/IP service and run it.

A TCP socket connection is made to the service and one line containing all the OLR data is sent to that process. The process should return one line containing the User ID for that particular end user or an error response.

The following protocol is used between the OLR server process and your UID Server:

- The process is sent all of the OLR data fields as a string containing *name=value* pairs in the format:

```
name_a=value_a&name_b=value_b&.....  
name_n=value_n [&] [<CR>] <LF>
```

For example:

```
First Name=John&Last Name=Smith&Social Security  
Number=039270492 [&] [<CR>] <LF>
```

- The UID Server protocol is permitted three possible responses to *sgreg*:

1. A User ID:

```
<userid> [<CR>] <LF>
```

This response indicates acceptance of OLR for the user whose ID is designated by *userid*.

2. A new line, indicating denial of OLR:

```
<NL>
```

This response, though it may be for a variety of reasons, is typically attributable to the lack of a match for *userid*; it is reported by *sgreg* to the OLR Client as an error E132 (i.e., an unexpected response from the UID Server).

### 3. An administrator-provided error message:

`<SPACE>DENY : <SPACE><text>`

This response is for the reasons stipulated by the administrator in *text*. The *text* must be printable characters; it can include spaces, but is otherwise completely free-form. When received by `sgreg`, the message will appear to the user as error E133 `<text>`.

where: `<SPACE>` is one space  
`<CR>` is a carriage return  
`<LF>` is a line feed  
`<NL>` is a new line

## Manual Setup of an HTML Page for On-Line Registration

As administrator, you may want to create your own OLR Web page customized to your company's needs, instead of using the automatically configured Web registration form (`http://your.smartgate.domain:3845/OLR`) provided by V-ONE. If you do, you will need to manually create an HTML form for your OLR Web Page.

In order to perform OLR, the administrator must set up a Web server with an HTML registration form. The action of this form is to connect to `localhost` on port 4090, passing on its data as an HTTP POST. Included in these data are such items as the name of the Registration Server, registration public key information, user inputs (e.g., first name, last name, phone number), addresses of interim firewalls, OLR data encryption level, and desired token type.

As the data are included in a POST, they are easily tailored by the administrator. For instance, the administrator can place a hidden field or user input for the SmartGate group.

**NOTE:** The leading space and the space following the colon are both significant.

**NOTE:** If you are not using the Single Port Proxy, SmartGate's OLR Web page is `http://your.smartgate.domain:2090/30reg.html`.

## Web Server Configuration

Currently a Web server must be configured to register users with the SmartGate Server. This includes setting up an FTP or an HTTP link to allow users to download the package, but primarily involves the creation of an HTML page or Web-based form that prompts the user for the required inputs, includes hidden fields with which to configure administrative preferences, and connects the user to SmartPass on the OLR port to begin OLR. The following is a sample form shell:

```
<HTML>
<HEAD>
<TITLE>SmartPass On-line Registration</TITLE>
</HEAD>

<h3>SmartPass On-line Registration</h3>
<FORM METHOD="POST" ACTION="http://127.0.0.1:4090/sgreg" target="_top">
    .
    .
    { . . . User Inputs . . . }
    .
    .
<INPUT TYPE="submit" NAME="Submit" value="register">
</FORM>
</BODY>
</HTML>
```

The action ACTION="http://127.0.0.1:4090/sgreg" in this shell specifies that the POST data will be posted to localhost on port 4090 (the OLR port) along with the action /sgreg, which will initiate OLR.

### Specification of Mandatory OLR Parameters

The three mandatory parameters for OLR are the SmartGate Server address, the public key name for OLR, and the public key itself. These are all included in the POST form as hidden input values because they are the same for all registering users. The OLR Client is expecting a certain input name for each of these three parameters and it will render a failure (Error) message in the browser if one of them is not specified. The input names themselves are not case sensitive nor are the values unless otherwise specified.

The address of the SmartGate Server to be used for the token boot information is specified by the input name "ServerAddress" as follows:

```
<INPUT TYPE="hidden" NAME="ServerAddress" VALUE="206.205.193.125">
```

The actual value for "ServerAddress" may be an IP address or a DNS name of less than 64 characters.

OLR uses a public key to secure the exchange of an authentication key, which is used to secure the session. This key and its keyname must be supplied to the client as hidden values. Locate the file *keyname.cer* in the SmartGate Server's root directory. The actual unencoded key is the content of this file. The input name for keyname is "PubKeyName" and that for the key

itself "PubKeyVal." You can copy and paste the key information directly into your HTML form, but be careful not to change any of the key information in either file. These input names and their (case-sensitive) values are represented in the POST form as follows:

```
<INPUT TYPE="hidden" NAME="PubKeyName" VALUE="TEST01">
<INPUT TYPE="hidden" NAME="PubKeyVal" VALUE="MQNSQVcxBlRFU1QwMTF+MHww
DQYJKoZIhvcNAQEBBQADawAwaAJhAKkIgkKzLV4VpE5eeLSW2ffgqYFZQ6eYkx8tkH2tZI
Y6R6/vVxuASQ2LBjRfjp1j9VU1B/9wr25IdCWSwZY1TmFF/eClwi8lyH0O9MXHkNRMnLO
FIy8Yuk0fY54m2wIDAQABMQZWLU9ORTAxgH1y7K07+N7EUSxFrWV/bG6YKzbqpJLjl/
rJgI7LmOlN1vcFoJIfmOhv67Ns8oPTmjfo3tZkRTIW8ZhE4QDMDB63ghHIIM2IXXick0
CMi2FidrSBLDTCGVMRx50pRPR+6IqjUzsQpNmmHT9eC4GWSQMptqJQuM7cIa+RYsC+Vci">
```

## User Inputs

The POST form may prompt the user for up to ten user inputs. This information will be passed to the server as the user's personal information record to be stored in the file `sgreg usr` after registration. Each user input field should have an input name beginning with a (case-sensitive) "UR#", which indicates to the OLR client that this is user input. Here the # is an integer representing the type of data expected. This integer will permit the OLR client to perform error checking on the input. The acceptable numbers are:

- 0 Any input
- 2 Alphanumeric input
- 3 Numeric input
- 4 Phone number
- 5 Social Security number

The first 2 lines must be alphanumeric, otherwise there are no restrictions on what information is requested of the user and what type. However, all user inputs are mandatory, so empty input fields will cause the return of an error message in the browser window and terminate OLR.

The following is an example of three user inputs:

```
First Name: <INPUT TYPE="text" NAME="UR2First Name" SIZE="25">
Last Name: <INPUT TYPE="text" NAME="UR2Last Name" SIZE="25">
Email Address: <INPUT TYPE="text" NAME="UR0Email" SIZE="50">
```

Alphanumeric input is expected for "First Name" and "Last Name", and any type of input for "Email". Note that the "UR#" is stripped off before the data are sent to the server; these are merely control data for the OLR Client.

## Specification of Optional OLR Parameters

If your SmartGate Server uses a nonstandard port for OLR (the standard for Single Port Proxy is 3845, and for multiple port is 2090), then specify your port with the "ServerPort" input name:

```
<INPUT TYPE="hidden" NAME="ServerPort" VALUE="2090">
```

The encryption level of the OLR data being sent to the SmartGate Server may be increased from DES to 3DES by the insertion of the hidden field "OLR\_EncryptMethod". For example:

```
<INPUT TYPE="hidden" NAME="OLR_EncryptMethod" VALUE="3DES">
```

This option is only available with SmartGate 2.7 or later and SmartPass 3.4 or later. Previous versions of either client or server will default to DES encryption. No special license is needed to implement this feature.

You can register to a different interface on the SmartGate Server by using the "ServerNameTag" input name, which specifies that OLR should use this value rather than that for Domainname for the token boot information:

```
<INPUT TYPE="hidden" NAME="ServerNameTag" VALUE="InsideIP">
```

If some of your users are registering through a firewall, you can prompt them for the firewall address using the "UseFirewall" input name. This field will be ignored if left blank.

```
<INPUT NAME="UseFirewall">
```

If all of your users use the same firewall, you can make the "UseFirewall" input a hidden input and specify the correct value to save your users the trouble.

You can also specify into which SmartGate group your users register.

```
<INPUT TYPE="hidden" NAME="UsergroupName" VALUE="bronze">
```

This can be a user input, or you can make it a hidden input and provide different registration Web pages for different groups.

## Customer Branding

The OLR Client sends responses back to the client browser indicating either success or failure. These responses can be branded to include the following company-specific information:

```
<INPUT TYPE="hidden" NAME="CompanyName" VALUE="XYZ Corporation">
```

```
<INPUT TYPE="hidden" NAME="StreetAddress" VALUE="200 A. Street">
```

```
<INPUT TYPE="hidden" NAME="City" VALUE="Silver Spring,">
```

```
<INPUT TYPE="hidden" NAME="State" VALUE="VA">
```

```
<INPUT TYPE="hidden" NAME="ZipCode" VALUE="21015">
```

```
<INPUT TYPE="hidden" NAME="Country" VALUE="United States">
```

```
<INPUT TYPE="hidden" NAME="PhoneNumber" VALUE="(900) 976-0101">
```

You can also specify a Web page that will appear as a hot link in the response (the default is <http://www.v-one.com/>):

```
<INPUT TYPE="hidden" NAME="WebPage" VALUE="http://www.yourdomain/">
```

Finally, you can specify an e-mail address for technical support which will be hot linked in the response:

```
<INPUT TYPE="hidden" NAME="Email" VALUE="support@yourdomain">
```

## Creation of a Desktop Shortcut

SmartPass comes with a program called `vspstart.exe`. This program launches SmartPass and the default Web browser, changes the browser proxy settings to `localhost` as necessary, and directs the browser to a site that is specified with a `-h` command line option. On-Line Registration creates a desktop shortcut to `vpstart.exe` with a description that you supply as a specification of the optional form parameter `StartDesc`:

```
<INPUT TYPE="hidden" NAME="StartDesc" VALUE="Describe your shortcut">
```

You may also supply command line options for `vspstart.exe` with the form parameter `StartArgs`:

```
<INPUT TYPE="hidden" NAME="StartArgs" VALUE="-h www.yourdomain">
```

## Sample Form

In essence, SmartPass OLR is nothing more than an advanced CGI script. The following is an example of a completed OLR form:

```
<HTML>
<HEAD><TITLE>SmartPass On-Line Registration: Step 2</TITLE></HEAD>
<BODY TEXT="#000000" BGCOLOR="#FFFFFF">

<center><h2>XYZ Corporation OLR</h2></center>
<FORM METHOD="POST" ACTION="http://127.0.0.1:4090/sgreg" target="_top">

</CENTER>

<!-- specify ServerAddress, PubKeyName, PubKeyVal in the fields below -->
<INPUT TYPE="hidden" NAME="ServerAddress" VALUE="!!!! IP address of OLR server">
<INPUT TYPE="hidden" NAME="OLR_EncryptMethod" VALUE="3DES">
<INPUT TYPE="hidden" NAME="ServerPort" VALUE="2090">
<INPUT TYPE="hidden" NAME="PubKeyName" VALUE="!!!! keyname of the SmartGate server">
<INPUT TYPE="hidden" NAME="PubKeyVal" VALUE="!!!! content of the .cer
file for the keyname specified above, embedded cr and lf are OK">

<!-- the following fields will be shown to the user when registration completes -->
<INPUT TYPE="hidden" NAME="CompanyName" VALUE="!!!! XYZ Corporation">
<INPUT TYPE="hidden" NAME="WebPage" VALUE="!!!! http://www.xyz.com/">
<INPUT TYPE="hidden" NAME="StreetAddress" VALUE="!!!! 123 XYZ Street">
<INPUT TYPE="hidden" NAME="City" VALUE="!!!! Xyzville,">
<INPUT TYPE="hidden" NAME="State" VALUE="!!!! State">
<INPUT TYPE="hidden" NAME="ZipCode" VALUE="!!!! 12345">
<INPUT TYPE="hidden" NAME="Country" VALUE="!!!! Country">
<INPUT TYPE="hidden" NAME="PhoneNumber" VALUE="!!!! (800) 123-4567">
<INPUT TYPE="hidden" NAME="Email" VALUE="!!!! support@xyz.com">
```



<HR>

Fill in all of the fields in the registration form below  
and press the <b>register</b> button begin on-line registration.

<PRE>

<!-- specify below the fields listed in the "reginfo.dat" file.

The format of the NAME= field is important.

It starts with UR to indicate that it's a reginfo.dat field.

Note that UR is case-sensitive, it must be uppercase.

Next follows a number to indicate the field type

- 0 anything
- 2 alphanumeric
- 3 numeric
- 4 phone
- 5 social security number

Then comes the fieldname shown in reginfo.dat file. There are  
a couple of sample fields included here....

->

First Name: <INPUT TYPE="text" NAME="UR2First Name" SIZE="25" MAXLENGTH="50">  
Last Name: <INPUT TYPE="text" NAME="UR2Last Name" SIZE="25" MAXLENGTH="50">  
Soc. Sec. #: <INPUT TYPE="text" NAME="UR5Social SecurityNumber" SIZE="25" MAXLENGTH="50">  
Phone: <INPUT TYPE="text" NAME="UR4Phone" SIZE="25" MAXLENGTH="50">  
Email Address: <INPUT TYPE="text" NAME="UR0Email" SIZE="25" MAXLENGTH="50">

Member Group: <SELECT NAME="UsergroupName">

<option>Bronze  
<option>Silver  
<option>Gold  
</select>

</PRE>

<p>

<center>

<table>

<tr>

<td>

<INPUT TYPE="submit" NAME="Submit" VALUE="register">

</td>

<td>

<INPUT TYPE="reset" NAME="Reset" VALUE="reset">

</td>

</tr>

</table>

</center>

</FORM>

</BODY>

` </HTML>

# SmartPass Deployability

Sometimes SmartPass needs to navigate a firewall because of a firewall security policy or corporate architecture, it cannot pass traffic. The `PortList=` option located in the `SmartGate sgconf.ini` and the `SmartPass setup.ini` files addresses this problem by enabling the SmartGate administrator to list the port(s) that the SmartPass software will use to attempt the firewall navigation for the first On-Line Registration.

`PortList=values`

This option is commented out (in both the `sgconf.ini` and `SmartPass setup.ini` files) with the default port values being 3845, 443, and 80 separated by commas. The administrator needs to uncomment the option and verify or insert which ports the SmartPass installation will use to try to navigate through the 'foreign' firewall.

During the first OLR, the `PortList` is consulted, in order. The first successful connection will stop the search and the valid port is written into the SmartPass registry. This port is used for all future connections.

Any change in `PortList` on the Server (`sgconf.ini`) after installation requires the administrator to manually stop and restart the `sgproxy` service.

## Microsoft Windows NT:

1. `sgproxy -unregserver`
2. `sgproxy -regserver`
3. In **Start, Settings, Control Panel, Services** stop the **SmartGate Server**.
4. In **Start, Settings, Control Panel, Services** start the **SmartGate Server**.

## Unix-based:

1. `ps ax | grep sgproxy`      BSD/OS, Linux  
   `ps -ef | grep sgproxy`      Solaris
2. `kill pid`      where pid is the process id of the process found in step 1
3. `/usr/smartgate/libexec/sgproxy -daemon`

Any change in `PortList` on SmartPass (`setup.ini`) after installation requires the end user to reinstall their SmartPass.

**WARNING!** If the deployability option is utilized, OLR MUST immediately follow the installation because the ports are written into memory!

**WARNING!** Do NOT use the default port settings of 443 or 80 if either an SSL Server or Web Server, respectively, is running on your SmartGate Server.

**NOTE:** The line length for `PortList` is 1000 characters maximum.

**WARNING!** Since the 'successful' port is written into the registry, all connections to additional SmartGate Servers must be made on the same 'successful' port.

**NOTE:** On Windows NT, steps 1 and 2 are required; steps 3 and 4 can be replaced by a reboot.

**NOTE:** The administrator can also reboot instead of manually stopping and restarting the `sgproxy` service.

**WARNING!** The Winsock shim does not work if a proxy is being used to traverse a firewall.

**NOTE:** For detailed descriptions of proxy options, see the “SmartPass Proxy Options” in Chapter 3, “Installing and Registering SmartPass for Microsoft Windows,” of the *SmartPass Administrator’s Guide*.

**WARNING!** If you are using the HTTP (Web or SSL tunneling) Proxy with authentication required, do not use automatic On-Line Registration. (Do not enter a Web OLR URL for the OLRPage option in the setup.ini file, located on your SmartPass installation disk.)

## Performing OLR Through a Firewall

If SmartPass is using a proxy to traverse an intermediate firewall, end users will need to set the address and port number of the proxy during or prior to performing OLR, depending on whether authentication is required. As administrator, you must instruct them on how to configure their SmartPass software. These steps must be performed by end users:

1. Install and then launch the SmartPass software.

The Web browser should automatically be configured to localhost (127.0.0.1).

2. Configure SmartPass to proxy through an intermediate firewall. If the proxy requires authentication, then it must be done prior to performing OLR.

You may configure a proxy:

- During OLR on the Web page, but only if the proxy does not require authentication.
- By opening the Control Panel Applet of the appropriate authentication method and selecting the **Settings** tab. The options are the same as for the **SmartPass Options** display as described below.
- By using the **SmartPass Options** display as described below.

Open the SmartPass User Interface from the Windows taskbar tray. Open the **SmartPass Options** display by clicking the **Options** button on the toolbar. Select the tab for the type of proxy being used.

- **Connect through proxy server**

Select from the **Proxy server** drop-down box **Connect through proxy server**. Enter the IP address or DNS name of your proxy server in the large text box and the port in the small text box.

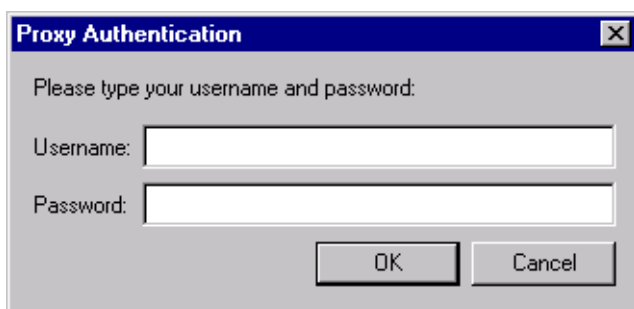
- **Connect through proxy server (authentication required)**

A firewall can be configured to require a username/password authentication every time a user attempts to traverse the firewall using a Web or SSL Proxy. SmartPass can be configured to prompt end users at the beginning of each SmartPass session for their firewall username and password. SmartPass will then pass the necessary information on to the firewall every time the proxy is

activated during a single SmartPass session. As administrator, you must assign each end user a username and password in accordance with your firewall software. If you are using multiple firewalls with authentication, the same username and password must be used for each proxy.

Select from the **Proxy server** drop-down box **Connect through proxy server (authentication required)**. Only the Web or the SSL proxy offer this option. Enter the IP address or DNS name of your proxy server in the large text box and the port in the small text box.

You will be prompted to enter a username and password (Figure 7-1). These codes must be entered every time SmartPass is opened and from this point on, every time you launch SmartPass you must enter your firewall username and password in the Proxy Authentication Dialog Box.

A screenshot of a Windows-style dialog box titled "Proxy Authentication". The dialog box has a blue title bar with a close button (X) in the top right corner. The main area is light gray and contains the text "Please type your username and password:". Below this text are two white text input fields. The first field is labeled "Username:" and the second field is labeled "Password:". At the bottom of the dialog box, there are two buttons: "OK" and "Cancel".

**Figure 7-1**  
**Proxy Authentication Dialog Box**

### 3. Perform On-Line Registration.

Click the Windows **Start** button. Select **Programs, V-ONE SmartPass 4.x**, and then **On-Line Registration**. Your Internet browser should be launched and a Web OLR form will be displayed. Enter the required data on the Web registration form and click **Register**.

**NOTE:** The URL for the OLR form is: `http://your.smartgate.domain:n/OLR`.

# Chapter 8

# Request for Passive Open (PASV)

**NOTE:** In most cases, PASV is used to route through firewall restrictions.

## PASV File Transfer Protocol

The PASSIVE command is defined in the RFC959 standard as follows:

This command requests the FTP Server to listen on a data port (which is not its default data port) and to wait for a connection rather than initiate one upon receipt of a transfer command. The response to this command includes the host and port address this server is listening on.

## Using the FTP Passive Mode With SmartGate

The FTP Passive mode (PASV) is a function of the application being used. SmartGate implementation of the RFC959 standard offers two ways in which users can take advantage of PASV.

### ■ Using an FTP client

Some FTP client applications can be manually set for PASV. Check the documentation prepared by the vendor of your FTP client to verify whether it supports PASV.

### ■ Using a Web browser

Some browsers support PASV. Check the documentation prepared by the vendor of your browser to verify whether it supports PASV.

## Setting Up an FTP Client That Supports PASV

To set up an FTP client using PASV, you must:

1. Ensure that the user has the necessary permissions for FTP access.

2. Provide your users with the following general instruction:
  - Start SmartPass, at which time they will receive their FTP access permissions.
  - Start their FTP clients.
  - If you are not using the shim, change the settings in their FTP clients to invoke PASV and set the FTP session to connect to `localhost` or `127.0.0.1`. (Your users should refer to the documentation prepared by the vendors of their FTP clients to complete these steps.)

# Chapter

# 9

**Macintosh USERS:**  
Multithreaded Oracle is not supported by SmartPass for Macintosh.

**NOTE:** IPSec is not supported when NAT is used. You must add a traditional (non-IPSec) access permission for Oracle.

## SmartGate Oracle SQLNet II Proxy

There are two modes of operation for the SmartGate SQLNet Proxy: single-threaded and multithreaded. Whether you are using Oracle's single-threaded service or multi-threaded service you will use SmartGate's `sgora` proxy. Instructions are detailed below.

## Setting Up the SmartGate Server for the Oracle SQLNet Proxy

The Oracle SQLNet Proxy is implemented in the `sgora` proxy.

### Windows NT:

The `sgora` proxy is installed and registered by default. No further setup on the server is required.

### UNIX:

Before you set up your Oracle SQLNet software, you should:

1. Ensure that the `sgora` proxy is located in the SmartGate Server `root/libexec` directory and is set up to listen to port 3521.
2. Ensure that your `/etc/services` file contains the line:  
`sgora 3521/TCP`
3. Ensure that your `startup` file contains the line:  
`/SmartGate Server's root directory/libexec/  
sgora -daemon sgora`
4. Verify that the `sgora` proxy is running by issuing the following command:

<code>ps ax grep sgora</code>	<b>BSD/OS or RedHat Linux</b>
<code>ps -ef grep sgora</code>	<b>Solaris</b>

## Removing or Reinstalling the Oracle Service

If you want to remove or reinstall the Oracle service for any reason, use the following instructions:

### For Windows NT:

1. Stop the SmartGate Service (Go to **Control Panel, Services, SmartGate Server**, and click **Stop**).
2. Open a command prompt in the SmartGate Server's root directory.
3. To install services, enter the command:  
**servers -regserver sgora**  
To remove services, enter the command:  
**servers -unregserver sgora**
4. Start the SmartGate Service (Go to **Control Panel, Services, SmartGate Server**, and click **Start**).

## Oracle Setup

In order to set up your Oracle SQLNet Client, you will have to:

1. Set up the Oracle SQLNet Client (install your software) on your computer (running Windows 95, Windows 98, or Windows NT).
2. Set up an SQLNet connect path using Oracle's SQLNet Easy Configuration. If you are using the shim, set the corresponding host address to the Oracle Server's IP address or the hostname. If you are not using the shim, the corresponding host address must be set to 127.0.0.1 to allow the Oracle SQLNet Client to connect to SmartPass.
3. Run Oracle's **Easy Configuration** from the Window's Program window.
  - Add a data base alias new service. This new service name can be any name you wish.
  - Select TCP/IP as the network protocol.
  - On the **Choose TCP/IP Host Name and Data Base** screen, enter 127.0.0.1 (or localhost) or the Oracle's Server host name that you want to connect to. This is the same host name that you enter when setting up the access permission.

**NOTE:** On Microsoft Windows NT (using the shim or not) the corresponding host address must be set to 127.0.0.1 if you are using Oracle 8.x or above.

**NOTE:** If you are not using Oracle's SQL Easy Configuration, a manual change can be made in the TNSNAMES file.



## Setting Up Access Permissions

Oracle SQLNet Proxy access permissions can be set up and managed through SmartAdmin. To add an Oracle access permission:

1. Open SmartAdmin and click the **TCP Access** tab.
2. Click the **Add** button

To create an ACL for single-threaded Oracle servers,

dest server:	(IP address or DNS name of server)
dest port:	1521 (usually)
server port:	2023
client port:	1521 (usually)

Do not use the “Oracle” choice in the Generic TCP window for single-threaded mode. You should choose **Other**.

To create an ACL for multi-threaded Oracle servers:

dest server:	(IP address or DNS name of server)
dest port:	1521 (usually)
server port:	3521
client port:	1521 (usually)

You can use **Oracle** in the Generic TCP box.

For more information on managing TCP access permissions, see “[TCP Access Permissions](#)” in Chapter 5, “Using SmartAdmin.”

## Testing Your Connection

Once you have set up your Oracle pathways, test your Oracle SQLNet connection as described below.

1. Install SQLPLUS (if you did not install it earlier) according to the instructions provided in your Oracle documentation.
2. Run SQLPLUS. You will be prompted for:
  - A User ID and password. Enter your Oracle database User ID and password in these fields.
  - A host string. Enter the same service name you used in step 2 of Oracle Setup.
3. Launch SQLPlus.
4. After you enter the host string, your Oracle SQLNet Client will connect to SmartPass, which will connect to the SmartGate Server.

**NOTE:** The host string is the new service name designated in the Easy Configuration program.

5. Once the connection is established, you will see the `SQL>` prompt. To test the connection, you could enter a simple database query and look for some data to be returned. For example:

```
select * from all_tables;
```

and press `ENTER`.

## Single Port Configuration

The following displays how Oracle is preconfigured in `sgproxy.conf`.

With the Winsock shim:

```
sgora3521 TCP Oracle Server 3521 #Secure Oracle
```

Without the Winsock shim:

```
sgora3521 TCP 127.0.0.1 3521 #Secure Oracle
```

## Troubleshooting Oracle

- On Oracle 8.x Servers for Solaris, if you attempt to make a connection above your multi-threaded user-limit, the server will start creating connections in dedicated mode. Dedicated mode is similar to single-threaded connections. SmartGate does not support dedicated mode, since it will be expecting a multi-threaded connection going to the Oracle Server. Workaround: increase the user-limit on the Oracle Server.
- All Oracle connections, whether it is single-threaded or multi-threaded, or if it is shim or `localhost` mode, communicate over the standard SmartGate port (usually 3845) only.
- Oracle client 8.x for Windows NT requires connecting to `localhost` (127.0.0.1) since it bypasses the shim. Oracle client 7.x on Microsoft Windows NT and 8.x for Windows 95/98 can point directly to the server. Pointing the server to `localhost` can be done via the `TSANAMES` file or via the Oracle Easy-Config application.
- Oracle Nameservers are not supported.
- The Oracle DataBase Ping utility always makes a single-threaded connection, so it is not compatible with the Oracle multi-threaded proxy.

**NOTE:** You must type the semicolon at the end of the line to indicate the end of the command.

# Chapter 10

## Using the vplug Proxy

The vplug Proxy provides virtual hosting for generic TCP and Web connections on a SmartGate Server. This pass-through proxy allows incoming connections to a single port (configured for multiple IP addresses) to be routed to multiple application servers according to the incoming IP address.

The syntax for starting the vplug Proxy is:

### Microsoft Windows NT

```
vplug -regserver  
vplug
```

### UNIX-based

```
vplug [[-v] | [-c] | [-p] | [-d port]] [-f cfname] [-n rname]
```

vplug must be restarted after a system reboot.

## How vplug Works

The vplug Proxy is an application-level proxy that provides configurable access control and logging features. vplug is capable of virtual hosting; if you configure a network interface with multiple IP addresses for a single interface, vplug can determine which application server to connect to according to which IP address received the connection. Multiple port applications, such as FTP and Oracle SQLNet are not ordinarily supported by vplug, however, SmartGate's Single Port Proxy (default 3845) routes all TCP and Web connections from a single port to a single port. As long as you are using the Single Port Proxy all TCP connections are supported by vplug.

**NOTE:** The Single Port Proxy is the SmartGate default. It is available with SmartGate 2.5 and SmartPass 3.2 and later versions.

The vplug Proxy reads its configuration from the configuration file, `netaccess.cf`, only at startup. However, a UNIX operating system will also respond to a **SIGHUP**. The default configuration file is located in:

#### Microsoft Windows NT

C:\SmartGate Server's root directory\data

#### UNIX-based

/SmartGate Server's root directory/etc

The vplug Proxy may be used to proxy NNTP, HTTP, or other TCP-based application requests (at the application level) using rules you supply. It is protocol-independent, so you can proxy a variety of other TCP-based applications. You can configure vplug to allow connections based on:

- Source IP address
- Source hostname
- Destination IP address
- Destination hostname
- Destination port

## Configuring the `netaccess.cf` File

vplug reads its rules from `netaccess.cf`. Each rule must begin with the name of the proxy (i.e., vplug) followed immediately by a colon, and must appear on a single line.

Below is a single `netaccess.cf` rule with only mandatory parameters displayed:

```
vplug: -virtdst 206.205.74.245 -virtport 389 -realdst 20.0.0.41
```

## Example of a `netaccess.cf` File

If you want to set up more than one address on a single interface you will need to add an alias to the network adaptor card.

#### Microsoft Windows NT:

1. Double-click the **Network** property icon from the **Control Panel**.
2. Click the **Protocol** tab.
3. Select **TCP/IP** and click **Properties**.
4. Click the **Advanced** button.
5. Add IP addresses by clicking the **Add** button in the IP Addresses section.

**NOTE:** If `netaccess.cf` is not there, vplug will not start.

**NOTE:** On a UNIX platform, vplug must be either restarted or sent a **SIGHUP**, every time the `netaccess.cf` or the `net_list` files are changed.

**NOTE:** On UNIX, you may use a different file name by entering the file name using the **-f** command line option.

**NOTE:** On UNIX, the name of the proxy may be changed using the **-n** option.

6. Enter the desired IP address under IP Address and the netmask under Subnet Mask (i.e., 1.1.1.1 and 255.255.255.255); and then click **Add**.

Repeat steps 5 and 6 for each additional address.

### UNIX-based:

Use the `aliases` option of `ifconfig`. For example:

```
ifconfig ed0 inet alias 1.1.1.1 255.255.255.255
```

```
ifconfig ed0 inet alias 1.1.1.2 255.255.255.255
```

sets up two aliases on the interface **ed0** (presumed to be configured). See `ifconfig(8)` for more information.

### Additional Routes

You must also add routes matching these aliases as follows:

1. Open a command prompt.
2. Type the following command for each alias:

```
route add          IP address 127.0.0.1
```

where: *IP address* is the IP address of the new alias.

For example:

```
route add          1.1.1.1 127.0.0.1
```

```
route add          1.1.1.2 127.0.0.1
```

Using the above example (two aliases added at 1.1.1.1 and 1.1.1.2), if you were to run `vplug` with the following options:

```
vplug -d http
```

and the following rules were in your `netaccess.cf` file:

```
vplug: -timeout 600
```

```
vplug: -virtdst 1.1.1.1 -virtport http -realdst www1  
-realport http
```

```
vplug: -virtdst 1.1.1.2 -virtport http -realdst www2  
-realport 80 -srcaddr *.mil
```

any connections to 1.1.1.1 on port 80 would automatically be connected to `www1` on that port, and any connections on 1.1.1.2 on port 80 would be connected to `www2` only if the client's address were in the `.mil` domain. All connections will timeout after 600 seconds of inactivity.

### netaccess.cf Parameters Supported by vplug

Table 10-1 lists the mandatory and optional parameters supported by `vplug`.

Format	Description	O/M	Default
-nsmismatch	Disconnects the client if the hostname returned from a reverse address lookup does not resolve to the address of the client. It will not disconnect the client if there is no DNS information for the client at all. This check is for probable spoofing attacks, but may cause problems if the client's DNS Server is not properly set up, especially if the client is a dual-homed computer.	O	N/A
-realdst <b>host</b>	Specifies the IP address or hostname to which vplug should connect.	M	
-realport <b>port</b>	Specifies the port to which vplug will connect on the ultimate destination.	O	
-reqdns	Disconnects the client if DNS information cannot be obtained for the client's IP address.	O	N/A
-srcaddr <b>net_list</b>	Allows connections only from a host whose name or address matches one of the host patterns in <i>net_list</i> .	O	Any
-timeout <b>sec</b>	Specifies that after <i>sec</i> seconds of inactivity, vplug will close the connection. This option must be on a line by itself (i.e., it may not be mixed with other configuration parameters) and applies to all vplug connections running under the same name.	O	300 seconds
-virtdst <b>host</b>	Specifies the IP address or hostname on which vplug will accept connections.	M	Any
-virtport <b>port</b>	Specifies the port on which vplug will accept connections.	M	

**Table 10-1**  
**vplug Mandatory and Optional Parameters in *netaccess.cf***

**KEY:**    **M** = Mandatory  
          **O** = Optional

**NOTE:** *net\_list* can be either a host address or the name of a file which contains a list of hosts. Information regarding the *net\_list* option is detailed in "[net\\_list Guidelines](#)."

**NOTE:** To define a *net\_list* file, place a “@” directly before the file name.

## *net\_list* Guidelines

The `-srcaddr net_list` option is either:

1. A host(s), designated directly in `netaccess.cf`, from which connections are allowed, for example:

```
vplug: -virtdst 206.205.74.245 -virtport 389  
-realdst 20.0.0.41 -srcaddr 206.20.20.100
```

or

2. A @ sign plus the name of a file containing a list of hosts from which connections are allowed, for example:

```
vplug: -virtdst 206.205.74.245 -virtport 389  
-realdst 20.0.0.41 -srcaddr @trusted-nets
```

A host/network pattern may be an IP address or hostname. It may also contain a \* wildcard (i.e., \*.ops.net) or use an *IP\_address:netmask* to match a single host, subnet, or network (e.g., 10.0.0.0:255.0.0.0).

## Using a *net\_list* File

The following are guidelines for creating a *net\_list* file.

- A *net\_list* file contains one or more host/network patterns.
- You can nest *net\_list* files within *net\_list* files by beginning the file name with an @ sign. This file will be opened as well, and any additional host patterns will be read. Therefore, if @additional\_hosts is in a trusted\_nets file, the file additional\_hosts will be opened, and any line not starting with a # will be read as one or more host or network patterns.
- vplug will try to open the filename as is for full paths and, if that fails, will try to open the filename relative to the SmartGate Server's root directory.

# vpplug Options for UNIX

You may use only one of the following four primary command line options (Table 10–2), **-c**, **-p**, **-d**, and **-v**.

Format	Description	O/M	Default
<b>-d port</b>	Starts vpplug as a daemon monitoring the <b>port</b> specified.	M	N/A
<b>-v</b>	Prints vpplug version information and then exits.	O	N/A
<b>-p</b>	Prints rules for <code>netaccess.cf</code> , and then exits.	O	N/A
<b>-c</b>	Prints general rules for analyzing <code>netaccess.cf</code> .	O	N/A

The following options (Table 10–3) may be used in conjunction with the **-c**, **-p**, or **-d** primary command line options. Use each options separately with the primary command.

Table 10–2  
vpplug Primary Commands



Table 10-3  
Additional vplug Options

Format	Description	O/M	Default
<b>-f cfname</b>	Changes the name of the configuration file from which vplug reads its rules. This option is useful in conjunction with the <b>-c</b> or <b>-p</b> options to test new configurations before implementing them.	O	/SmartGate Server's root directory/netaccess.cf
<b>-n rname</b>	Changes the name under which vplug will run. vplug will report to syslog and get its rules from the configuration file under the specified name. The name may be up to 56 characters.	O	vplug

**NOTE:** On Windows NT, vplug needs to be unregistered and then reregistered every time the netaccess.cf or the net\_list files are changed.

## vplug for Windows NT

1. Create the netaccess.cf file and save in:  
C:\SmartGate Server's root directory\data
2. Stop the SmartGate Service (Go to **Control Panel, Services, SmartGate Service**, and click **Stop**).
3. Open a DOS command window.
4. Navigate to the SmartGate directory.
5. To register the vplug proxy, type:  
**vplug -regserver**
6. To start the vplug service, type:  
**vplug**  
The vplug service must be started after each reboot.
7. Start the SmartGate Service (Go to **Control Panel, Services, SmartGate Service**, and click **Start**).
8. To remove the vplug proxy service, type:  
**vplug -unregserver**



# Chapter 11

## IPSec

**NOTE:** IPSec is only available with SmartGate 4.0 and later running on a Microsoft Windows NT Server.

**NOTE:** For an updated list of IPSec-compatible network adaptor cards, see: <http://www.v-one.com/techsupport/tn001.htm/>.

### What is IPSec?

The set of protocols known as IPSec are defined in RFCs 2401-2412. There is also an older version of the standard described in RFCs 1825-1829. V-ONE's IPSec supports both sets of standards, but it is assumed that the older standards will only be used in cases where compatibility with an older existing IPSec implementation is required.

### IPSec's Core Components

There are two parts to the SmartGate IPSec protocol suite:

1. **ESP – Encapsulating Security Payload**
  - Provides the encryption part of IPSec
2. **AH – Authentication Header**
  - Provides a packet-by-packet authentication of the entire packet, including the IP headers
  - Can be applied alone, but usually done in conjunction with ESP
  - It does not work in situations where intermediate gateways on the network perform NAT
    - NAT modifies the IP header, which will render the hash incorrect and cause the packet to fail authentication

IPSec works at the network transport layer, below IP. Unlike traditional SmartGate, which deals with streams of data above the level of TCP, IPSec deals with packets of data from the stream after IP has already packetized it.

## Transport and Tunnel Mode

### Transport Mode

- Used for end-to-end communication between two hosts
- The payload is just the data, not the IP header, and only the data is protected
- Best for small networks where each host has IPSec

### Tunnel Mode

- Used when either end is a security gateway with IPSec (router, firewall, VPN gateway)
- The payload is the end user's entire packet (IP headers and data), and it is placed inside another packet—no routers are able to see the inner IP header
- The new packet has different source and destination addresses, thus providing greater security

## ESP - Encapsulating Security Payload

ESP provides the encryption part of IPSec, as well as optionally calculating a packet-by-packet authentication hash of the payload of each packet (but not the IP headers).

V-ONE supports DES and 3DES encryption types in ESP, and SHA1 and MD5 authentication hashes. The older versions of DES and 3DES encryption are also supported, including both 32-bit and 64-bit initialization vectors, but these should be avoided, since the newer ESP protocol has replay-protection (each packet has a sequence number that cannot be repeated, so the same packet can't be sent over and over again by an attacker).

## AH - Authentication Header

AH provides a packet-by-packet authentication of the entire packet, including the IP headers. Although the AH authentication can be considered more secure than ESP, it does not work in situations where intermediate gateways on the network perform NAT on the authenticated packets—NAT modifies the IP header, which will render the hash incorrect and cause the packet to fail the authentication.

V-ONE's AH supports MD5-HMAC and SHA-HMAC authentication hashes. Basically, the MD5 is faster, and SHA is cryptographically stronger.

**NOTE:** SmartGate always uses tunnel mode.

## IPCOMP - IP Payload Compression

IPCOMP is a method of compressing the data in the payload of an IP packet so that it will take up less bits on the wire. V-ONE supports only the DEFLATE protocol (which uses the same deflate algorithm used by the popular UNIX `gzip` file compression utility).

When compression is selected, an attempt is made to compress each packet over about 100 bytes long. If the resulting packet is no shorter, the original packet is sent, otherwise the compressed packet is sent.

## NAT - Network Address Translation

NAT translates the IP addresses on a packet as it passes a gateway (in this case, the SmartGate Server). NAT has many uses, the most common being to allow a large number of hosts to share a single IP address (another name for this use is NAPT (Network Address and Port Translation) and 1:N NAT). V-ONE does NOT use NAT for this purpose. V-ONE's NAT is a "1:1" NAT, and we use it for two reasons:

1. Providing each remote PC with a known IP address on the protected network.
2. Allowing the SmartGate Server to be located on the network "off to the side" from the natural route from the protected servers back to the clients on the Internet.

Currently, SmartGate only NATs the client's address, and each client gets a unique address while it is connected. This eliminates many (but not all) of the problems associated with 1:N NAT (where all clients would share the same IP address on the protected network). For example, if 1:N NAT was used, it wouldn't be possible for others to mount the drives on remote clients, but this is possible in our system.

An important note: currently the NAT address for each client is assigned automatically from a pool of addresses when the client authenticates and does DNR. This address remains allocated to the given client not just until the client disconnects, but until the server is restarted. This means that the administrator must allocate a pool of addresses large enough to have a unique address for every user (not just every simultaneous user). Since this address pool can be from one of the reserved private address spaces (10.0.0.0, 192.450.0.0, etc), this shouldn't be a problem.

## Putting Together ESP, AH, IPCOMP, and NAT

The easiest way to explain how all these protocols fit together is to describe the path of a packet as it travels from a remote client to its intended server on the private network.

When an application on the client wants to send data to another machine, it calls Winsock with some data. Winsock in turn passes this data down to the Microsoft Windows IP module, which breaks the data up into packet-size chunks, adds an IP header (containing information about source and destination addresses, protocol, port, length, etc.) and sends it down to the next lower layer. Normally, the next layer is the driver for the network adapter card, but when V-ONE's IPSec is installed, the packet is instead sent to the IPSec driver.

IPSec compares the packet to an access control list (ACL) that it was given during dynamic configuration and determines whether or not the packet should be encapsulated (tunneled, encrypted, etc.). If not, the packet is forwarded down to the adapter driver and continues. If the policy does show that the packet should be encapsulated, the following steps are done (the original packet is "IP"):

1. A new IP header is prepended to the packet, with a destination of the remote gateway (the SmartGate Server), and an indicator that this header must be stripped at the gateway before forwarding the packet on to its original destination (this is called IP in IP encapsulation/tunneling):

IPINIP (IP)

2. If IPCOMP is indicated, the packet is compressed, and a new header put on to indicate that it should be decompressed at the gateway on the other end:

IPCOMP (IPINIP (IP) )

It is very important to do the compression now, before the data is encrypted, since encrypted data is effectively random bits, and random bits don't compress well at all.

3. If ESP is indicated, the packet is encrypted with the given keys, and again a header indicating that decryption is needed is added to the packet:

`ESP (IPCOMP (IPINIP (IP) ) )`

4. If AH is indicated, a hash calculation is done on the packet, and the result is stored in a header along with the indicator that hash verification is required at the other end:

`AH (ESP (IPCOMP (IPINIP (IP) ) ) )`

Now the packet is sent to the gateway (SmartGate Server), where the reverse operation is done. After the packet is decapsulated at the server, a check is made in the server's policy database to see if this packet is actually allowed onto the private network. If it is, the client's address is replaced with its "private" NAT address (if the policy indicates) and the packet (now back in its original form) is forwarded on to the originally intended server.

The return packet follows much the same path, except that the client's address (which is now the destination address of the packet) is "UN-NATed" right after checking policy on the server, before any compression or encryption is done.

Usually, all the traffic between a client and server uses the same "tunnel" (encryption type, keys, compression, etc.). If a particular type of data doesn't need the same level of security (e.g., if there is a Usenet news server on the private network that only contains public groups), SmartGate can be configured using SmartAdmin to use a different "channel type" for that data, and a second tunnel (channel) will be setup between the gateway and each client.

# IPSec Functionality

SmartGate 4.0 for Windows NT and SmartPass 4.0 for Windows 95, 98, and NT include driver-level IPSec transport functionality. IPSec is a method of encapsulating IP packets (not just TCP sessions) to obscure (encrypt) and protect the data from modification enroute.

V-ONE's implementation also includes support for:

- RFC-standard IP packet payload compression, which compresses the packets before they are encrypted, therefore increasing throughput.
- Network Address Translation (NAT).
- A packet filtering engine to implement policy on the traffic flowing in and out of both the client and server.

The IPSec feature provides the following capabilities:

- **Additional protocol support**

IPSec can carry **any** IP traffic—TCP, UDP, ICMP, or any other IP protocol. In particular, this means that remote users can now share disks and printers using MS networking, and “browse” Microsoft Windows Network Neighborhood.

- **IPSec can block all traffic that it is not securing**

- **Improved performance for dial-up connections**

IPSec throughput is improved over older versions of SmartPass because of the IP payload compression feature. Although already compressed data streams (e.g., transferring a ZIP file) do not benefit, regular text (e.g., HTML, text documents, uncompressed TIFF files) can benefit considerably.

- **Fully compatible with existing SmartGate**

A single SmartGate Server can serve traditional SmartGate access permissions simultaneously with IPSec access permissions (even to the same client), so upgrading is a smooth process. SmartGate 4.0 will support users who have older versions of SmartPass—versions that don't support IPSec—and users can continue to use traditional SmartGate in situations that warrant it (e.g., when setting up a



connection that traverses multiple NAT gateways/firewalls). This is true of the client side as well—a client can still connect simultaneously to multiple servers, with each server providing a mix of traditional and IPSec access permissions.

## Choosing Your IPSec Network Topology

SmartGate with IPSec has three possible network configurations:

1. **Parallel** (Figure 11-1)

SmartGate with two interfaces placed in parallel with firewall (external interface on DMZ, internal interface on private network).

Encrypted traffic goes through SmartGate, non-encrypted traffic through the firewall.

2. **Serial** (Figure 11-2)

SmartGate with two interfaces placed directly behind the firewall, in serial.

All traffic passes through both the firewall and SmartGate.

3. **Toaster** (Figure 11-3)

SmartGate with a single interface placed anywhere on the internal network.

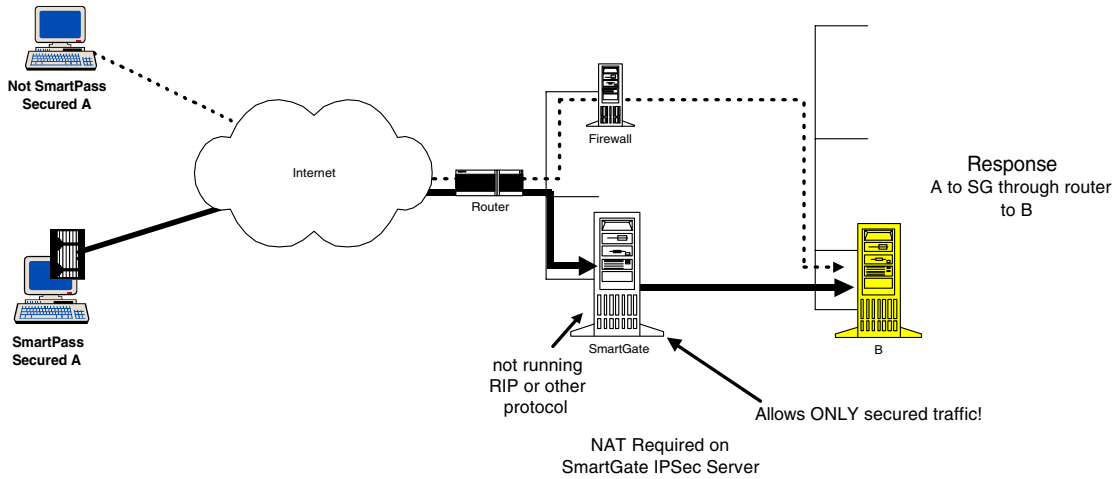
**NOTE:** The topology recommended for best performance is “parallel.”

**NOTE:** NAT is required for “parallel” and “toaster.” It is not required for “serial,” but can be used.

**NOTE:** You can have more than one “internal” and/or “external” interface, as long as all routing issues are handled properly by the network infrastructure.

**NOTE:** Serial and parallel topologies require 2 network adaptor cards, however, toaster only requires 1.

## PARALLEL w/ NAT

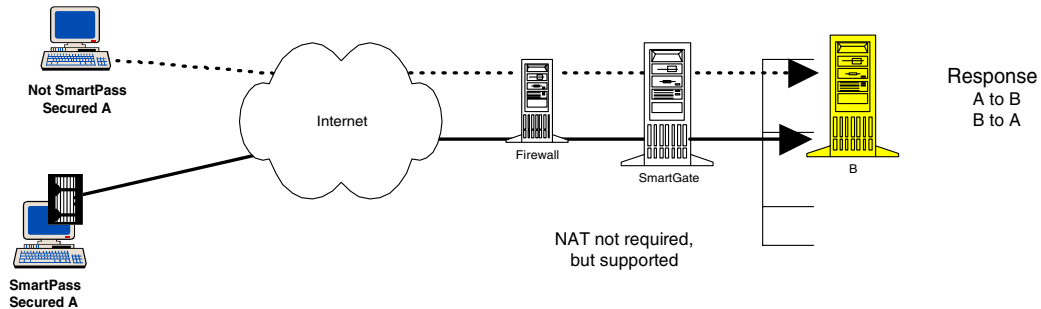


The dotted line is unsecured, allowed only if the firewall allows

The solid line is SECURED, allowed only if SmartGate allows

**Figure 11-1. SmartGate System Network Configuration: Parallel with NAT Mode**

## SERIAL

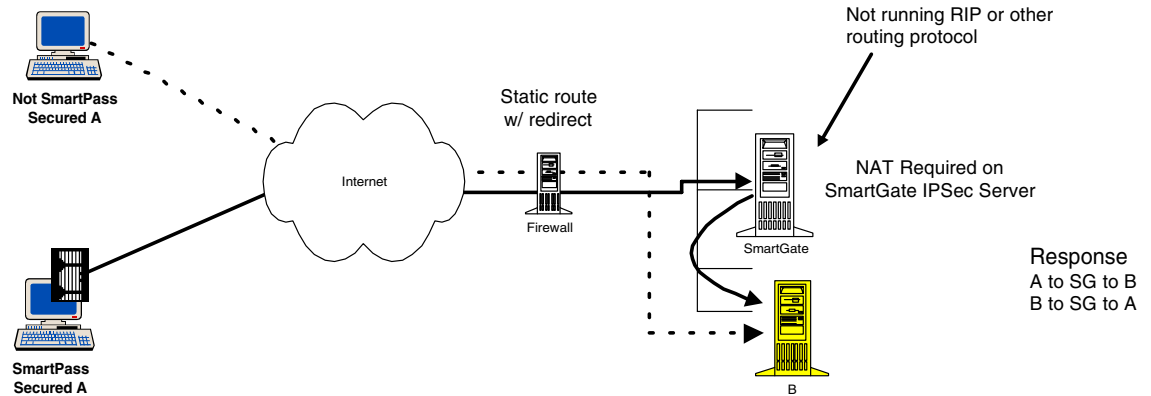


The dotted line is unsecured, allowed only if the firewall allows

The solid line is SECURED, allowed only if SmartGate allows

**Figure 11-2. SmartGate System Network Configuration: Serial Mode**

## TOASTER w/NAT



The dotted line is unsecured, allowed only if the firewall allows

The solid line is SECURED, allowed only if SmartGate allows

*Figure 11-3. SmartGate System Network Configuration: Toaster with NAT Mode*

## Routing

The biggest hurdle to be overcome by new users of SmartGate with IPsec is routing. The one exception to this is when using the “serial” topology with no NAT. In all other cases, routing must be configured in the following way (the term “NAT subnet” refers to the block of addresses allocated to the SmartGate NAT pool, which really should be an entire subnet):

1. All machines on the private net except SmartGate should have a route to the NAT subnet that points to the internal interface of the SmartGate Server.
2. SmartGate should have a route to the NAT subnet that points to the next hop towards the Internet after SmartGate’s external interface.

Because SmartGate must have a different idea of the route to the NAT subnet than everyone else, it is almost a necessity to NOT be running a routing protocol daemon on the SmartGate machine—it should use static routes.

**NOTE:** If a “larger” route pointing in the right direction already exists in SmartGate’s routing table, you do not need to add a specific route.

## Example:

If you have a SmartGate Server configured in the “parallel” topology, the internal interface is at address “I.I.I.I”, the external interface is at “E.E.E.E”, and the router connecting to the Internet is address “R.R.R.R”; SmartGate is configured to give NAT addresses from the 192.168.10.0/24 network.

What to do:

1. On the SmartGate Server, your default route (i.e., “default gateway”) probably already points to R.R.R.R, therefore, you don’t need to do anything. However, if for some reason your default route does not point to R.R.R.R, you need to add the following route:

```
destination: 192.168.10.0 mask 255.255.255.0
gateway: R.R.R.R
```

2. For the other machines on the network, add the following route in a strategic place (e.g., on the firewall, or on a router that broadcasts routing information to all other routers on the network):

```
destination: 192.168.10.0 mask 255.255.255.0
gateway: I.I.I.I
```

Another example:

You have a SmartGate Server configured in “toaster” mode with interface address “T.T.T.T”, and the firewall’s internal interface is R.R.R.R. Again, you will be using the 192.168.10.0/24 network for your NAT net.

What to do:

1. On the SmartGate Server, your default route to the NAT net probably already points to R.R.R.R, therefore, you don’t need to do anything. However, if for some reason your default route does not point to R.R.R.R, you need to add the following route:

```
destination: 192.168.10.0 mask 255.255.255.0
gateway: R.R.R.R
```

2. For other machines on the network, add a route for 192.168.10.0 pointing to SmartGate’s single interface:

```
destination: 192.168.10.0 mask 255.255.255.0
gateway T.T.T.T
```

**NOTE:** In STRICT mode, even 'ping' packets are not allowed, and no response will be given to them.

## Security Levels for Adapters (Interfaces)

Each interface on a SmartGate Server can be set to one of 4 security levels using SmartAdmin:

- **OFF**—The IPSec driver is disabled on this interface, and it behaves just as it would without SmartGate installed (i.e., it passes all packets in both directions without modification).
- **ON**—The IPSec driver is enabled, but unencrypted packets are also allowed through the interface.
- **ON-NOREDIRECT**—Just like ON, but ICMP redirect packets sent by Windows NT are discarded before they can leave the machine.
- **STRICT**—only encrypted traffic is allowed in and out this interface. ALL other traffic is blocked.

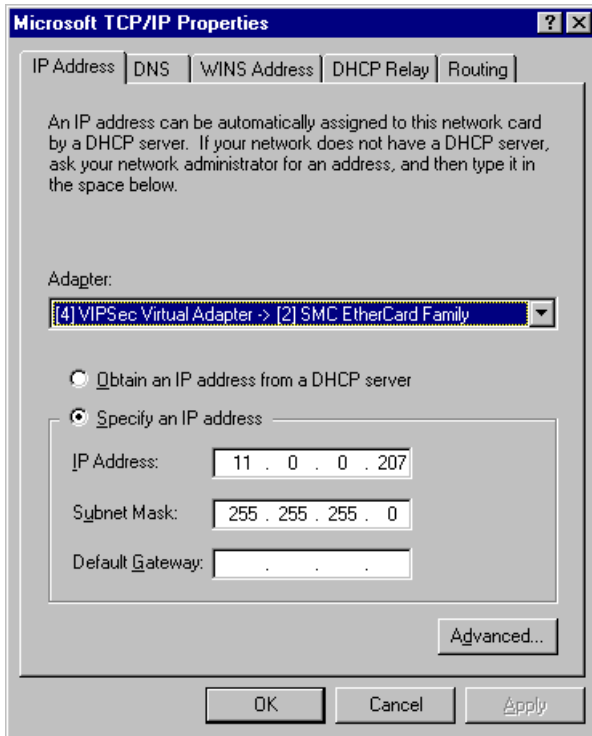
## Setting Up Your TCP/IP Protocol Properties

V-ONE recommends that all network TCP/IP Protocol properties be established according to your network configuration on your server prior to installing the SmartGate 4.x Server software.

TCP/IP properties include:

- Specify an IP address and Subnet Mask
- Specify the Default Gateway
- Specify the WINS Servers addresses
- Specify the DNS Servers
- Enable Routing—If IPSec is being used, IP forwarding MUST be turned on

To access the TCP/IP protocol properties on a Microsoft Windows NT Server, double-click **Network** in the Windows Control Panel. Click the **Protocol** tab, select **TCP/IP**, and click **Properties**. An example of the IP Address—TCP/IP Protocol Properties Window is displayed in Figure 11–4.



**Figure 11-4**  
**TCP/IP Protocol Properties**  
**Window**

On a Microsoft Windows NT Server, if you add or remove any adapter or protocol, you must remove and reinstall the IPsec driver. This can be done by removing or reinstalling the smartGate Server software or by opening an MS/DOS command prompt and typing:

**vipsecin /remove**

You will be prompted to reboot. Then type:

**vipsecin /install**

Again, you will be prompted to reboot.

**WARNING!** Removing any adapters while the IPsec driver is installed is not supported.

**NOTE:** For more information on SmartAdmin, see [Chapter 5](#), “Using SmartAdmin.”

**Figure 11-5**  
*IPSec Settings Window*

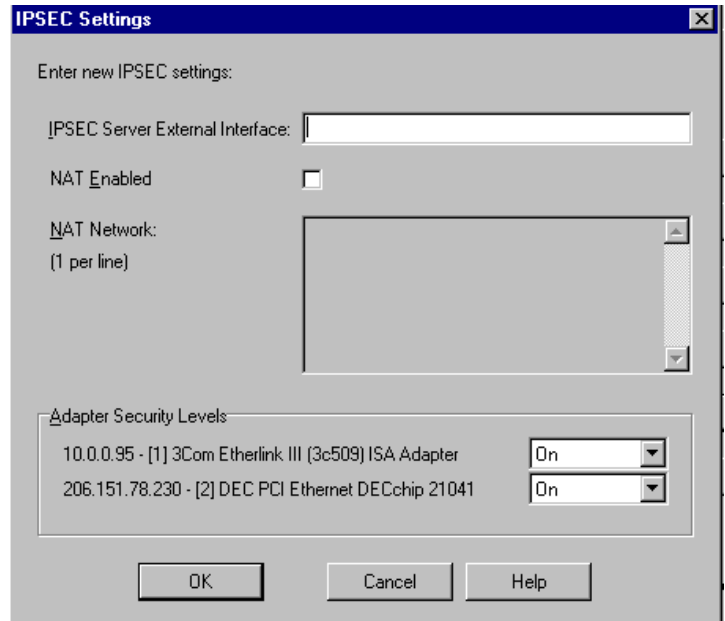
**NOTE:** IPSec Settings Window reflects information found in the `sgconf.ini` file located in SmartGate Server’s root directory\data on a Microsoft Windows NT.

**NOTE:** If you use AH, at no point from the client to the Server, can NAT be used. If you need to use AH, you must define a new channel type using ESP Authentication.

## Configuring IPSec Settings

The SmartGate Administrative GUI, SmartAdmin, should be used to configure all IPSec settings, channel types, and access permissions.

To open the IPSec Settings Window (Figure 11-5), click **IPSec** on the Configuration Window.

The image shows a screenshot of the 'IPSEC Settings' window. The title bar is blue with the text 'IPSEC Settings' and a close button. The main area is light gray. At the top, it says 'Enter new IPSEC settings:'. Below this is a text box labeled 'IPSEC Server External Interface:'. Underneath is a checkbox labeled 'NAT Enabled', which is currently unchecked. Below the checkbox is a text box labeled 'NAT Network:' with '(1 per line)' below it. At the bottom, there is a section titled 'Adapter Security Levels' containing two entries: '10.0.0.95 - [1] 3Com Etherlink III (3c509) ISA Adapter' and '206.151.78.230 - [2] DEC PCI Ethernet DECchip 21041'. Each entry has a dropdown menu next to it, both currently set to 'On'. At the bottom of the window are three buttons: 'OK', 'Cancel', and 'Help'.

### IPSec Server External Interface ([ipsec\\_server\\_extrn](#))

Specifies the external interface (IP address) where you want IPSec tunnels to be terminated. Normally this would be the external interface of the SmartGate Server itself. However, if the SmartGate Server is behind a proxy firewall or a packet filtering firewall that is doing NAT, you may have to set the IPSec Server External Interface to the address of the external interface of the firewall instead.

### NAT Enabled (NAT)

Specifies whether or not the IPSec Server should perform Network Address Translation (NAT) on encrypted traffic. NAT is disabled by default (i.e., the checkbox is not selected). If NAT is enabled, you must specify a range of addresses to be used for address translation in the **NAT Network** setting.

## NAT Network (NATNet)

If the NAT Enabled checkbox is selected, specify the address range that the IPSec Server can use for address translation. The address range or list of networks should be formatted as either **IP:mask** or **IP\sum of masks**. For example:

**10.10.100:255.255.255.0**

or

**10.10.10.0\24**

## Adapter Security Levels

The adapter security level defines the strictness of the security on a particular adapter using the following options:

- OFF:** The internal interface(s) of a multiple interface SmartGate should be set to OFF. All packets are allowed in and out of the adapter without being encrypted (i.e., it passes all packets in both directions without modification).
- ON:** “Permissive.” All packets are allowed in and out, but they are checked and encrypted or decrypted if applicable. This mode is useful in the serial configuration.
- ON-NOREDIRECT:** Exactly like ON, except that ICMP redirect messages originating on the SmartGate Server are not allowed to leave the machine. This is useful in a toaster configuration, where cleartext packets come in one interface, are encrypted, and leave by the same interface.
- STRICT:** Only packets that are encrypted, including those on TCP port 3845, are allowed to go in and out of the adapter. All other packets are dropped or rejected. This mode is most useful in the parallel configuration.

In general, this is how you should configure your interfaces:

### ■ Serial topology:

External interface = ON

Internal interface = OFF

Because all traffic coming in from the firewall must also go through the SmartGate Server, and some of that traffic may not be encrypted, you need to allow it to pass. (The firewall has already done a security screening of the non-encrypted traffic.)

**NOTE:** The Adapter Security Levels setting is stored in the registry, not `sgconf.ini`.

**NOTE:** Although this setting affects outbound ICMP redirect packets, it does not affect ICMP redirect packets sent to the SmartGate Server by other machines on the network.

**NOTE:** In STRICT mode, even ‘ping’ packets are not allowed, and no response will be given to them.

**NOTE:** A scan of an interface in STRICT mode should show that TCP port 3845 (used for Dynamic Configuration and traffic in proxy mode) is open, and everything else is blocked.



## ■ Parallel topology:

External interface = STRICT

Internal interface = OFF

Since the external interface is connected directly to the Internet (effectively), you want to make sure nothing gets in/out besides encrypted traffic matching the IPSec ACLs set in SmartAdmin.

## ■ Toaster mode:

External/Internal interface = ON-NOREDIRECT

The packets sent to the SmartGate are redirected back out the same interface by NT's IP, which also sends out an "ICMP redirect" packet to the sender, informing it of a "more efficient" path to the real host.

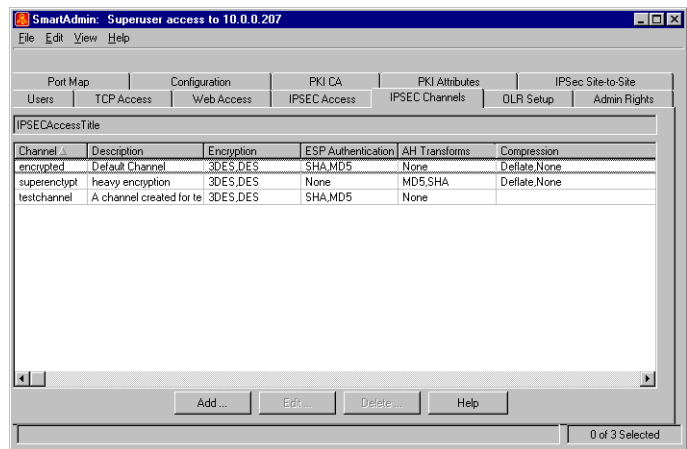
## Configuring IPSec Channel Types

The IPSec Channels Table (Figure 11–6) allows you to configure the IPSec channels which are used when creating IPSec access permissions. The "encrypted" IPSec channel type is installed by default as displayed below with the SmartGate Server.

**Figure 11–6**  
**IPSec Channels Table**

**NOTE:** The IPSec Channels Table reflects information found in the `chantype.ini` file located in the SmartGate Server's root directory\data.

**NOTE:** The default encrypted channel should not be deleted.



Channel	Description	Encryption	ESP Authentication	AH Transforms	Compression
encrypted	Default Channel	3DES,DES	SHA,MD5	None	Deflate, None
superencrypt	heavy encryption	3DES,DES	None	MD5,SHA	Deflate, None
testchannel	A channel created for te	3DES,DES	SHA,MD5	None	Deflate, None

### 1. Channel

Specifies the name of the channel. You designate the channel name; however, it must be alphanumeric without spaces or special characters and with a maximum of 16 characters.

### 2. Description (optional)

An optional text description of the channel.

3. **Encryption** Specifies acceptable [ESP](#) encryption methods in order of preference. The value string is displayed as a comma-delineated list of available methods; the preferred method will be listed first. Available methods include: 3DES, DES, and None.
4. **ESP Authentication** Specifies acceptable ESP authentication methods in order of preference. The value string is displayed as a comma-delineated list of available methods; the preferred method will be listed first. Available methods include: SHA, MD5, and None.
5. **AH Transforms** Specifies acceptable [AH](#) authentication methods in order of preference. The value string is displayed as a comma-delineated list of available methods; the preferred method will be listed first. Available methods include: SHA, MD5, and None.
6. **Compression** Specifies acceptable compression methods in order of preference. Available methods include: Deflate and None.

When managing IPSec channels, three commands are available:

- Add channel
- Edit channel
- Delete channel

**NOTE:** Generally, MD5 authentication is better for speed and SHA authentication is more cryptographically strong (RFC 2104).

## Add/Edit IPsec Channels

To add or edit IPsec channels, use the command buttons at the bottom of the window or the menu bar options. Figure 11-7 is an example of the Add IPsec Channel Window. The Add IPsec Channel Window contains the same information in a nearly identical layout.

**Figure 11-7**  
**Edit IPsec Channel Window**

The screenshot shows a dialog box titled "Edit IPSEC Access Permission". It contains the following elements:

- Edit IPSEC channel definition:**
  - Channel name:** A text box containing "test".
  - Description:** A text box containing "just a test".
- ESP Encryption:** A group box containing three radio buttons: ☒ 3DES, ☐ None, and ☐ DES.
- ESP Authentication:** A group box containing three radio buttons: ☒ SHA, ☐ None, and ☐ MD5.
- Authentication Header:** A group box containing three radio buttons: ☒ None, ☐ MD5, and ☐ SHA.
- Compression:** A group box containing two radio buttons: ☒ Deflate and ☐ None.
- Buttons:** At the bottom are three buttons: "OK", "Cancel", and "Help".

As administrator, the Channel name and Description titles are your choice, simply type in the text you want to use. Creating a description is optional.

Using the definition on the previous page, select ESP Encryption methods for this channel. For authentication methods you must choose between ESP and AH authentication. **You cannot use both for the same channel.** Select None for the type of authentication you are NOT using.

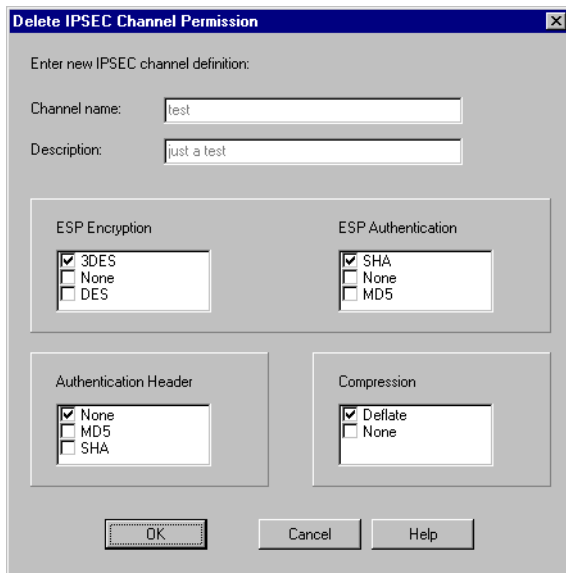
For compression, if you are communicating over a dial-up, it is advisable to use the deflate compression option. However, if you are communicating over Ethernet to Ethernet with fast processors, then you must consider total performance. In terms of the amount of time it takes to send a packet, four things need to be considered:

1. Wire time
2. OS overhead (x2)
3. Encryption/decryption (x2)
4. Compression/decompression (x2)

After selecting which types you want available for that specific IPSec channel, you need to put them in order of preference. To do this, select the type (it will be highlighted) and use the up and down arrows to move your selection higher or lower in your priority list.

## Delete IPSec Channels

To delete an IPSec channel, select a single channel or multiple channels and either click **Delete** at the bottom of the window or use the menu bar option. Figure 11-8 is displayed.



The image shows a Windows-style dialog box titled "Delete IPSEC Channel Permission". It contains the following elements:

- Title Bar:** "Delete IPSEC Channel Permission" with a close button (X).
- Text:** "Enter new IPSEC channel definition:"
- Channel name:** A text box containing "test".
- Description:** A text box containing "just a test".
- ESP Encryption:** A group box containing three radio buttons: ☒ 3DES, ☐ None, and ☐ DES.
- ESP Authentication:** A group box containing three radio buttons: ☒ SHA, ☐ None, and ☐ MD5.
- Authentication Header:** A group box containing three radio buttons: ☒ None, ☐ MD5, and ☐ SHA.
- Compression:** A group box containing two radio buttons: ☒ Deflate and ☐ None.
- Buttons:** "OK", "Cancel", and "Help" buttons at the bottom.

**Figure 11-8**  
*Delete IPSec Channel Window*

The Delete IPSec Channel Window is a confirmation that you want to delete that channel. If you select multiple channels for deletion, the system will ask for a deletion confirmation on each channel.

## Configuring IPsec Access Permissions

The IPsec Access Table (Figure 11–9) displays the access records of all IPsec access permissions.

**Figure 11–9**  
**IPsec Access Table**

**NOTE:** IPsec is only available with SmartGate 4.0 and later running on a Microsoft Windows NT server.

**NOTE:** The IPsec Access permissions table reflects information found in the `ipsec.acl` file located in the SmartGate Server's root directory\data.

Owner Type	Owner ID	Perm. Type	Destination Host / Group	Destination network mask	P
Group	Sales	Path	10.0.0.122	255.240.0.0	tc
Group	Sales	DNS Proxy	10.0.0.1	255.255.255.255	
Group	Finance	Include	ProjectX		
Group	Management	Path	10.0.0.1	255.255.255.255	a

Buttons: Add..., Edit..., Delete..., Help

0 of 4 Selected

Each IPsec access record (row) on the IPsec Access Table displays a single permission assigned to either a group or an individual user. Each record contains between three and eight fields, depending on the permission type.

When managing IPsec access permissions, three commands are available:

- Add an access permission
- Edit an access permission
- Delete access permissions

## Add/Edit IPSec Path or Include Access Permissions

To add or edit IPSec access permissions, use the command buttons at the bottom of the window or the menu bar options. Figure 11–10 is an example of the Add IPSec Path Access Permission Window. The Edit IPSec Path Access Permission Window contains the same information in a nearly identical layout.

The screenshot shows a Windows-style dialog box titled "Add IPSEC Access Permission". Inside, there's a section "Enter new IPSEC access permission:". Below this, there are two radio buttons for "Owner": "User" and "Group", with "Group" selected. Next to it is a text field for "Owner ID" containing "Sales". Below that is a section for "Type" with three radio buttons: "Path permission" (selected), "DNS Proxy", and "Include group of permissions". Further down are two text fields: "Destination host" with "10.0.0.122" and "Destination mask" with "255.240.0.0". Below these are four dropdown menus: "Protocol" set to "tcp", "Port" set to "1", "Priority" set to "Medium", and "Channel" set to "encrypted". At the bottom are three buttons: "OK", "Cancel", and "Help".

**Figure 11–10**  
**Add IPSec Path Access**  
**Permission Window**

**NOTE:** For a complete listing of the protocol definitions see [“Protocol Number Definitions”](#) later in this chapter.

The following list defines options available when the Permission Type is Path permission or Include group of permissions. The DNS Proxy Permission Type is defined in the following section.

1. **Owner** Specifies whether this permission is assigned to a user or a group.
2. **Owner ID** Either the User ID, the name of the group that will receive this permission, or “all.” All users belong to the universal group “all.”



3. **Type**

The value for this field can be:

**Path**—indicates that a path to a remote host and access permission has been designated.

**DNS Proxy**—is a DNS Proxy statement. If the DNS Proxy type is chosen, the remaining fields will appear differently. Please refer to Figure 11–11.

**Include**—indicates that the specified Owner (either User or Group) will acquire the paths of the included group.

Individual path permissions display in black, DNS Proxy statements display in blue, and included group permissions display in green.

4. **Destination Host**

The value for this field depends on what the Permission Type is as follows:

**Path**—the hostname or IP address of the path's destination host.

**Include**—the name of the included group.

5. **Destination mask**

**Path (only)**—a subnet mask on Destination Host.

6. **Protocol (Optional)**

**Path (only)**—Specifies the IP protocol to which this path applies. The default IP protocol is **all**, specifying all possible IP protocols. You may define this path to a single protocol such as: TCP, UDP, ICMP, or any of the other available communication IP protocols.

7. **Port**

**Path (only)**—This field is only valid when Protocol is ICMP, TCP, or UDP.

TCP or UDP—it is the remote port of the path. A wildcard port “\*” is acceptable. Leaving the field blank defaults to “\*”.

ICMP—the field changes to **Type**, which refers to the ICMP type as defined by the ICMP RFC 792.

- |                    |  |
|--------------------|--|
| 8. <b>Priority</b> | Lowest to highest.   |
| 9. <b>Channel</b>  | <p>A channel selected from the channels defined on the IPSEC Channels tab or “BLOCK” which will cause the path to be blocked.</p> <p>“encrypted” is the only default channel. Any additional channels must be defined using SmartAdmin’s Channels tab.</p> |

## Delete IPsec Path Permission

To delete an IPsec access permission, highlight the permission and click delete. You will verify the deletion by clicking **Delete**.

## IPSec DNS Proxy Overview

Most people who have a private network also have a private DNS. Some people advertise all their private addresses out to the Internet, but most don’t. Most people have some kind of firewall and their private network back behind it.

When a client (running SmartPass or not) connects to the Internet, he gets a DNS address from the ISP with a DNS server. When the client connects to ISP, the ISP says you should use this DNS server. If you are using that DNS server and you ask for something in `Company123.com`, (if it doesn’t know what the address is) it will forward the request to Company123’s public DNS server. It does this by asking the root server “who handles `Company123.com`” and the root server responds by saying “this DNS server does.” The problem is, `mail.company123.com` isn’t entered into the public DNS (because Company123 does not want this to be public information). So, when the request is forwarded to Company123’s public DNS server, the response is: “I don’t know and I am the authoritative source of information for the `Company123.com` domain, so there is no such place.” If the request was sent to Company123’s private DNS server, this server would know who `mail.company123.com` was, but the DNS has been configured by the ISP to point only to the Company123’s public DNS server.

One way to circumvent this problem is to go into the network configuration and enter the IP address of the private DNS server. You go into Control Panel/Network Setup/IP and go to DNS Servers and type in an address. Then, whenever you do a DNS lookup, Windows will first look to the DNS server you



**NOTE:** The DNS Proxy is a feature of SmartGate 4.0, which requires IPsec, but is not a part of the IPsec standard.

entered by hand. If that DNS server doesn't respond (this would happen if SmartPass was not running) the request would fail. This requires all users to manually configure their IP Network Setup.

Instead, what SmartGate does is just take whatever DNS is configured—no manual reconfiguration necessary. When clients run SmartPass and the IPsec driver is activated, one of the types of IPsec access permissions they can access is a DNS Proxy. "Proxy" is a bit of a misnomer since normally proxies take place at the application level, but V-ONE's IPsec, where the DNS proxying is performed, resides at the driver level of the IP stack. However, it performs the functions of a proxy.

### **Example:**

If the Domain=`*.v-one.com` and the DNS server is `192.43.000.4` (which in this example is the private DNS server), the DNS request goes down through the entire network stack (through the IP down to the IPsec driver) and it notices that this is a packet to UDP port 53 (i.e., a DNS request). It compares the name that's being asked for and if it matches with `*.v-one.com`, it modifies the packet so that instead of going to the ISP's address, it changes the destination address of the packet to go to `198.69.135.6`. It also puts it into the encrypted channel, which is this tunnel going back to the SmartGate Server. The SmartGate Server will decrypt the packet and forward it to the private DNS server. The private DNS server gets the answer and sends it back. You can have access to multiple networks and multiple (separate) private DNS servers. You just can't have conflicting address space.

## Add/Edit IPsec DNS Proxy Access Permissions

Figure 11-11 is an example of the Add IPsec DNS Proxy Access Permission Window. The Edit IPsec DNS Proxy Access Permission Window contains the same information in a nearly identical layout.

**Add IPSEC Access Permission**

Enter new IPSEC access permission:

**Owner:**  
☐ User      Owner ID:   
☒ Group

**Type:**  
☒ Path permission  
☒ DNS Proxy  
☐ Include group of permissions

Destination host:

Destination mask:

Protocol:       Port:

Priority:

Channel:

OK      Cancel      Help

**Figure 11-11**  
**Add IPsec DNS Proxy Access**  
**Permission Window**

Using the field descriptions provided, fill in the blanks with the appropriate information.

1. **Owner** Specifies whether this permission is assigned to a user or a group.
2. **Owner ID** Either the User ID, the name of the group that will receive this permission, or “all.” All users belong to the universal group “all.”
3. **Type** **DNS Proxy**—is a DNS Proxy statement.  
DNS Proxy statements display in blue.
4. **Domain Name** **DNS Proxy**—a wildcarded Domain Name. This is the Domain Name of the DNS Server to which your end

users should have access. This DNS Server will be used to resolve names for a particular domain or subdomain and will give permission for the DNS packets that perform this resolution.

5. **DNS Server**

**DNS Proxy**—the IP address of the DNS Server.

6. **Channel**

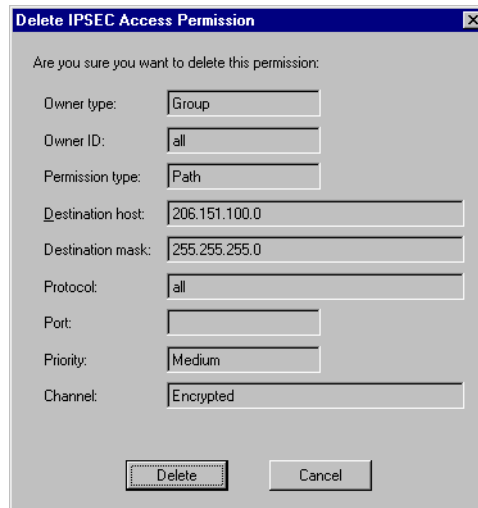
A channel selected from the channels defined on the IPSEC Channels tab. Do not set the channel to BLOCK.

“encrypted” is the only default channel. Any additional channels must be defined using SmartAdmin’s Channels tab.

## Delete IPsec Access Permissions

To delete IPsec access permissions, select an access permission record or multiple records and either click **Delete** at the bottom of the window or use the menu bar option. Figure 11–12 is an example of the Delete IPsec Group Access Permission Window.

**Figure 11–12**  
**Delete IPsec Access Permission Window**



The Delete IPsec Access Permission Window is a confirmation that you want to delete the permission. If you select multiple permissions for deletion, the system will ask for a deletion confirmation on each permission.

## Using the VIPUTIL Utility

VIPUTIL is normally not needed because the functionality it provides are available through SmartAdmin. Currently VIPUTIL allows you to:

1. View the Interface/Adapters Security Levels
  - VIPUTIL level  
Will display a list containing a numeric 'handle' for the interface and the security level for the adapter (OFF,ON,ON-NOREDIRECT,STRICT)
2. Set the Security Level for an Interface/Adapter
  - VIPUTIL level <handle> <SecurityLevel>
  - Example: VIPUTIL level 0 OFF (this means set adapter 0's level to OFF)
3. Log all packets on all interfaces that have a level setting higher than OFF:
  - VIPUTIL log
4. Log all packets on a particular interface (as long as its level is not OFF):
  - VIPUTIL log <handle>
  - Example: VIPUTIL log 0
5. Periodically (once a second, as long as there is traffic) print throughput stats for the interfaces whose level is not OFF. The throughput numbers printed are a running 5-second average, and list cleartext throughput separately from IPSec encapsulate" (encrypted/authenticated) traffic. Note that traffic running through a SmartGate Proxy is tallied in the cleartext category, since these stats are coming from the IPSec driver, which sees the (already encrypted by the proxy) traffic as being just cleartext that happens to be random.
  - VIPUTIL throughput
6. Periodically (any time there is traffic on an interface within a 1 second period) print out packet/byte totals for interfaces. These numbers show the total number of packets/bytes transferred through the interface since the machine was booted.
  - VIPUTIL stats

## Protocol Number Definitions

In the Internet Protocol version 4 (IPv4)[RFC791] there is a field, called “Protocol,” to identify the next level protocol. This is an 8-bit field. In Internet Protocol version 6 (IPv6) [RFC883] this field is called the “Next Header” field.

Dec.	Keyword	Protocol	Reference
0	HOPOPT	IPv6 Hop-by-Hop Option	[RFC1883]
1	ICMP	Internet Control Message	[RFC792]
2	IGMP	Internet Group Management	[RFC1112]
3	GGP	Gateway-to-Gateway	[RFC823]
4	IP	IP in IP (encapsulation)	[RFC2003]
5	ST	Stream	[RFC1190,IEN119]
6	TCP	Transmission Control	[RFC793]
7	CBT	CBT	[Ballardie]
8	EGP	Exterior Gateway Protocol	[RFC888,DLM1]
9	IGP	any private interior gateway (used by Cisco for their IGRP)	[IANA]
10	BBN-RCC-MON	BBN RCC Monitoring	[SGC]
11	NVP-II	Network Voice Protocol	[RFC741,SC3]
12	PUP	PUP	[PUP,XEROX]
13	ARGUS	ARGUS	[RWS4]
14	EMCON	EMCON	[BN7]
15	XNET	Cross Net Debugger	[IEN158,JFH2]
16	CHAOS	Chaos	[NC3]
17	UDP	User Datagram	[RFC768,JPB]
18	MUX	Multiplexing	[IEN90,JPB]
19	DCN-MEAS	DCN Measurement Subsystems	[DLM1]
20	HMP	Host Monitoring	[RFC869,RH6]
21	PRM	Packet Radio Measurement	[ZSU]
22	XNS-IDP	XEROX NS IDP	[ETHERNET,XEROX]
23	TRUNK-1	Trunk-1	[BWB6]
24	TRUNK-2	Trunk-2	[BWB6]
25	LEAF-1	Leaf-1	[BWB6]
26	LEAF-2	Leaf-2	[BWB6]
27	RDP	Reliable Data Protocol	[RFC908,RH6]
28	IRTP	Internet Reliable Transaction	[RFC938,TXM]
29	ISO-TP4	ISO Transport Protocol Class 4	[RFC905,RC77]
30	NETBLT	Bulk Data Transfer Protocol	[RFC969,DDC1]
31	MFE-NSP	MFE Network Services Protocol	[MFENET,BCH2]
32	MERIT-INP	MERIT Internodal Protocol	[HWB]
33	SEP	Sequential Exchange Protocol	[JC120]
34	3PC	Third Party Connect Protocol	[SAF3]
35	IDPR	Inter-Domain Policy Routing Protocol	[MXS1]
36	XTP	XTP	[GXC]
37	DDP	Datagram Delivery Protocol	[WXC]

38	IDPR-CMTP	IDPR Control Message Transport Proto	[MXS1]
39	TP++	TP++ Transport Protocol	[DXF]
40	IL	IL Transport Protocol	[Presotto]
41	IPv6	Ipv6	[Deering]
42	SDRP	Source Demand Routing Protocol	[DXE1]
43	IPv6-Route	Routing Header for IPv6	[Deering]
44	IPv6-Frag	Fragment Header for IPv6	[Deering]
45	IDRP	Inter-Domain Routing Protocol	[Sue Hares]
46	RSVP	Reservation Protocol	[Bob Braden]
47	GRE	General Routing Encapsulation	[Tony Li]
48	MHRP	Mobile Host Routing Protocol	[David Johnson]
49	BNA	BNA	[Gary Salamon]
50	ESP	Encap Security Payload for IPv6	[RFC1827]
51	AH	Authentication Header for IPv6	[RFC1826]
52	I-NLSP	Integrated Net Layer Security TUBA	[GLENN]
53	SWIPE	IP with Encryption	[J16]
54	NARP	NBMA Address Resolution Protocol	[RFC1735]
55	MOBILE	IP Mobility	[Perkins]
56	TLSP	Transport Layer Security Protocol using Kryptonnet key management	[Oberg]
57	SKIP	SKIP	[Markson]
58	IPv6-ICMP	ICMP for IPv6	[RFC1883]
59	IPv6-NoNxt	No Next Header for IPv6	[RFC1883]
60	IPv6-Opts	Destination Options for IPv6	[RFC1883]
61		any host internal protocol	[IANA]
62	CFTP	CFTP	[CFTP,HCF2]
63		any local network	[IANA]
64	SAT-EXPAK	SATNET and Backroom EXPAK	[SHB]
65	KRYPTOLAN	Kryptolan	[PXL1]
66	RVD	MIT Remote Virtual Disk Protocol	[MBG]
67	IPPC	Internet Pluribus Packet Core	[SHB]
68		any distributed file system	[IANA]
69	SAT-MON	SATNET Monitoring	[SHB]
70	VISA	VISA Protocol	[GXT1]
71	IPCV	Internet Packet Core Utility	[SHB]
72	CPNX	Computer Protocol Network Executive	[DXM2]
73	CPHB	Computer Protocol Heart Beat	[DXM2]
74	WSN	Wang Span Network	[VXD]
75	PVP	Packet Video Protocol	[SC3]
76	BR-SAT-MON	Backroom SATNET Monitoring	[SHB]
77	SUN-ND	SUN ND PROTOCOL-Temporary	[WM3]
78	WB-MON	WIDEBAND Monitoring	[SHB]
79	WB-EXPAK	WIDEBAND EXPAK	[SHB]
80	ISO-IP	ISO Internet Protocol	[MTR]
81	VMTP	VMTP	[DRC3]
82	SECURE-VMTP	SECURE-VMTP	[DRC3]
83	VINES	VINES	[BXH]
84	TTP	TTP	[JXS]
85	NSFNET-IGP	NSFNET-IGP	[HWB]

86	DGP	Dissimilar Gateway Protocol	[DGP,ML109]
87	TCF	TCF	[GAL5]
88	EIGRP	EIGRP	[CISCO,GXS]
89	OSPFIGP	OSPFIGP	[RFC1583,JTM4]
90	Sprite-RPC	Sprite RPC Protocol	[SPRITE,BXW]
91	LARP	Locus Address Resolution Protocol	[BXH]
92	MTP	Multicast Transport Protocol	[SXA]
93	AX.25	AX.25 Frames	[BK29]
94	IPIP	IP-within-IP Encapsulation Protocol	[JI6]
95	MICP	Mobile Internetworking Control Pro.	[JI6]
96	SCC-SP	Semaphore Communications Sec. Pro.	[HXX]
97	ETHERIP	Ethernet-within-IP Encapsulation	[RXH1]
98	ENCAP	Encapsulation Header	[RFC1241,RXB3]
99		any private encryption scheme	[IANA]
100	GMTP	GMTP	[RXB5]
101	IFMP	Ipsilon Flow Management Protocol	[Hinden]
102	PNNI	PNNI over IP	[Callon]
103	PIM	Protocol Independent Multicast	[Farinacci]
104	ARIS	ARIS	[Feldman]
105	SCPS	SCPS	[Durst]
106	QNX	QNX	[Hunter]
107	A/N	Active Networks	[Braden]
108	IPComp	IP Payload Compression Protocol	[RFC2393]
109	SNP	Sitara Networks Protocol	[Sridhar]
110	Compaq-Peer	Compaq Peer Protocol	[Volpe]
111	IPX-in-IP	IPX in IP	[Lee]
112	VRRP	Virtual Router Redundancy Protocol	[Hinden]
113	PGM	PGM Reliable Transport Protocol	[Speakman]
114		any 0-hop protocol	[IANA]
115	L2TP	Layer Two Tunneling Protocol	[Aboba]
116	DDX	D-II Data Exchange (DDX)	[Worley]
117	IATP	Interactive Agent Transfer Protocol	[Murphy]
118	STP	Schedule Transfer Protocol	[JMP]
119	SRP	SpectraLink Radio Protocol	[Hamilton]
120	UTI	UTI	[Lothberg]
121	SMP	Simple Message Protocol	[Ekblad]
122	SM	SM	[Crowcroft]
123	PTP	Performance Transparency Protocol	[Welzl]
124	ISIS over IPv4		[Przygienda]
125	FIRE		[Partridge]
126	CRTP	Combat Radio Transport Protocol	[Sautter]
127	CRUDP	Combat Radio User Datagram	[Sautter]
128	SSCOPMCE		[Waber]
129	IPLT		[Hollbach]
130	SPS	Secure Packet Shield	[McIntosh]
130-254		Unassigned	[IANA]
255		Reserved	[IANA]

## Site-To-Site IPSec

Site-to-Site IPSec enables the SmartGate administrator to set up VPN connections between entire networks via an IPSec tunnel from a Microsoft Windows NT SmartGate Server installed on one network to a Windows NT SmartGate Server installed on another. The parameters for SmartGate's Site-to-Site IPSec are:

- Networks connected by SmartGate's Site-to-Site must each have a unique IP address space.
- There is never any direct communication between the two SmartGate Servers—they are setup manually using SmartAdmin by the administrators at both ends. The only traffic between the Servers is traffic that is being forwarded.
- Although SmartGate's client/server VPN connections can have usage policies based on port and protocol, as well as server-side IP address, site-to-site VPN connection policies are based solely on source and destination IP addresses—port/protocol information cannot be used in deciding which traffic is allowed to flow between the sites.

## Operational Description

When the IPSec service is started (usually at boot), it will check for the presence of a file called `sites.acl` in the SmartGate directory. If this file is found, it will be read and interpreted by SmartGate, and the configuration for the local ends of IPSec tunnels for each listed remote site will be loaded into the IPSec driver. The same process occurs at the remote site.

Once both sites have (independently) initialized their drivers, all traffic entering the Servers will be monitored. Properly configured packets with a destination address of one of the remote networks will be encrypted and sent to the proper remote SmartGate Server for decryption and forwarding to the intended destination. Return packets will follow a similar path. The decision of which packets to send through which tunnel will be based on lists of local and remote hosts/networks designated as having permission to communicate via the tunnels.

## Firewall Considerations

Firewall considerations are the same as for client/server IPSec—any firewalls between the two SmartGate Servers must support passage of IPSec packets, and if any NATing is performed, the firewalls must properly NAT IPSec packets.

**NOTE:** Each network must have a single SmartGate that can send/receive IPSec (IP Protocol 50 and/or 51) packets to and from the Internet.

**NOTE:** SmartGate does not perform the Network Address Translation (NAT) that would be required to support connecting two networks which both use the same IP network addresses.

**NOTE:** Properly configured packets contain source and destination addresses which are allowed to traverse a particular tunnel.



**NOTE:** The requirements for setting up these routes is *\*exactly\** the same as the requirements for setting up the route to the NAT range used by SmartGate IPSec client-server communication—both could use exactly the same description.

**NOTE:** The IPSec Site-to-Site Configuration Tab is visible only when the Server has the IPSec capabilities installed.

## Routing Considerations

In order for traffic destined for a remote network to be encrypted, the traffic must be routed through the local SmartGate Server. If the SmartGate Server is on the natural path to the Internet (Serial configuration), this will happen by default. However, if the SmartGate Server is installed in Toaster or Parallel configuration, the administrator will need to add static routes for the remote networks that point towards the SmartGate Server.

For example, if the local private network is 10.0.0.0/24, the local SmartGate Server is 10.0.0.113, and the remote network is 192.168.2.0/24, the administrator must enter a route with the following parameters:

Destination: 192.168.2.0

Netmask: 255.255.255.0

Gateway: 10.0.0.113

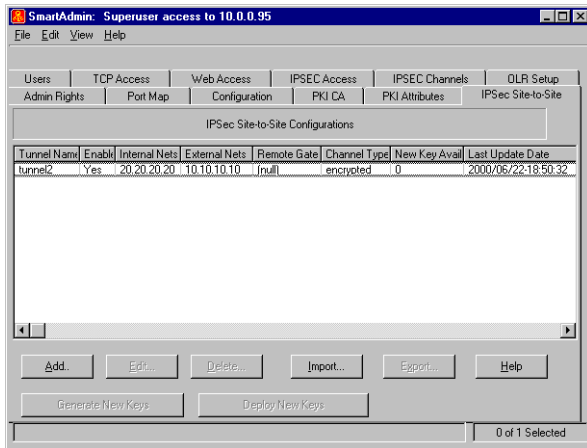
This route should be setup at least on the default gateway for the local network; adding it to other routers/hosts on the network may improve efficiency, or in some cases even be required, depending on network topology.

## Configuration Information

- A set of encryption keys must be sent from one site to another via a secure, out-of-band method (i.e., encrypted e-mail, diskette, etc.). Only static keys are supported (i.e., the keys will only change as a result of an explicit action by the administrator, followed by sending a new set of keys to the remote site.
- Configuration information ([sites.acl](#)) of the SmartGate Servers IPSec tunnel's local ends must be conveyed to each remote administrator via an out-of-band method. The format of this file is in [Appendix A](#). If new IPSec tunnels are created after boot time, then this information is written immediately to the IPSec driver.

## The Site-to-Site Tab

Site-to-Site configuration is performed using the IPSec Site-to-Site Tab in SmartAdmin, Figure 11–13. This table contains a list of all the currently configured tunnels to remote sites.



**Figure 11-13**  
**IPSec Site-to-Site Configuration Table**

**NOTE:** The IPSec Site-to-Site Configuration Table reflects information found in the `sites.acl` file located in the SmartGate Server's root directory\data.

The following information of each Site-to-Site IPSec tunnel is displayed.

<b>Tunnel Name</b>	Name of this site-to-site connection. NO spaces allowed.
<b>Enabled</b>	Yes or No. Enabled tunnels are displayed in black type. Disabled tunnels are displayed in red type, and tunnels that have new keys generated but have not been activated are displayed in blue type.
<b>Internal Nets</b>	List of network addresses on the local private network that will be accessible to the remote network via this tunnel.
<b>External Nets</b>	List of network addresses on the remote private network that will be accessible to the local network via this tunnel.
<b>Remote Gateway</b>	The other IPSec Server to use as the Server-side gateway for this tunnel.
<b>Channel Type</b>	The channel type to be used for this tunnel from the <code>chantype.ini</code> file.
<b>New Keys Available</b>	Designates whether new keys have been generated; but not deployed.
<b>Last Update Date</b>	The date the keys were last updated. Disabled tunnels display in red text. Tunnels that have new keys generated, but not activated, display in blue text.

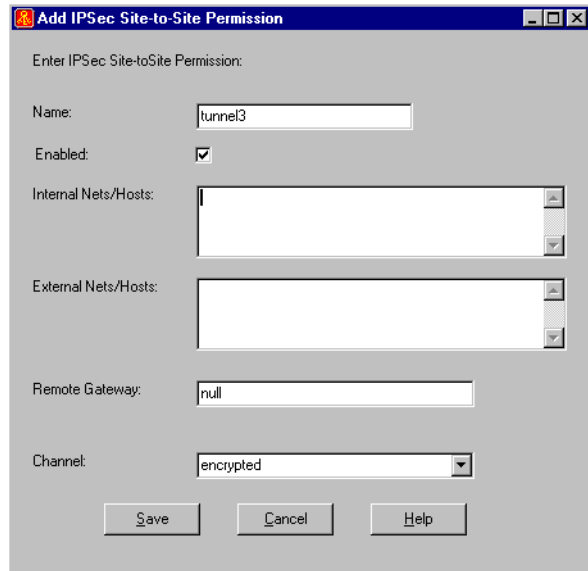
**Figure 11-14**  
**Add IPSec Site-to-Site Permission**  
**Window**

**NOTE:** Random keys based on the default channel generated during this **Add** function. By default, it uses the encrypted channel.

**WARNING!** Do not use a domain name for the internal and external nets, IP addresses are accepted only.

## Adding an IPSec Site-to-Site Permission

To add/edit an IPSec tunnel to SmartGate click the Add button on the IPSec Site-to-Site tab, Figure 11-14 is displayed.



Enter the following information for the new tunnel.

1. **Tunnel Name** can be a maximum of 1024 characters.
2. **Enabled**
3. **Internal Nets/Hosts** is a list of IP addresses of the local tunnel. Either separate the addresses by commas or place each address on a separate line.  
Type: ASCII string of:
  1. dotted quads (e.g., 1.2.3.4)
  2. dotted quads with netmask specifier (e.g., 192.168.2.0/24)Meaning: List of network addresses on the local private network that will be accessible to the remote network via this tunnel.  
Default: None - required field
4. **External Nets/Hosts** is a list of IP addresses of the remote tunnel. Either separate the addresses by commas or place each address on a separate line.  
Type: ASCII string of:
  1. dotted quads (e.g., 1.2.3.4)
  2. dotted quads with netmask specifier (e.g., 192.168.2.0/24)

Meaning: List of network addresses on the remote private network that will be accessible to the local network via this tunnel.

Default: None - required field

5. **Remote Gateway** is a single line IP address of the remote end of the IPsec tunnel.

Type: <dotted quad>

Meaning: Address of remote end of tunnel

Default: None - required field

6. **Channel** will be selected from the dropdown menu. The channel types coincide with the channels listed under the IPsec Channel Tab in SmartAdmin.

Type: ASCII string

Meaning: Name of a channel type to be used for this policy (from `chantype.ini`)

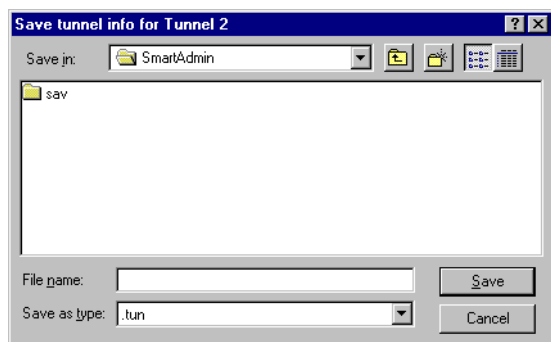
Default: "Default." Use the default channel type for this SmartGate server.

**NOTE:** By default, the channel is encrypted.

## Exporting A New Site-to-Site IPsec Tunnel

The Export button displays a Save to File dialog window. It will save all of the fields, including the [Site] at the beginning and the next . \* fields, but it will save them with tx and rx reversed (and also with intrn and extrn reversed). All selected tunnels will be saved, each to a separate file having the name TunnelName.tun.

1. Select the site's entry on the IPsec Site-to-Site Configuration Tab, Figure 11-13. Figure 11-15 is displayed.



**Figure 11-15**  
**Save Tunnel Information for Tunnel 2**

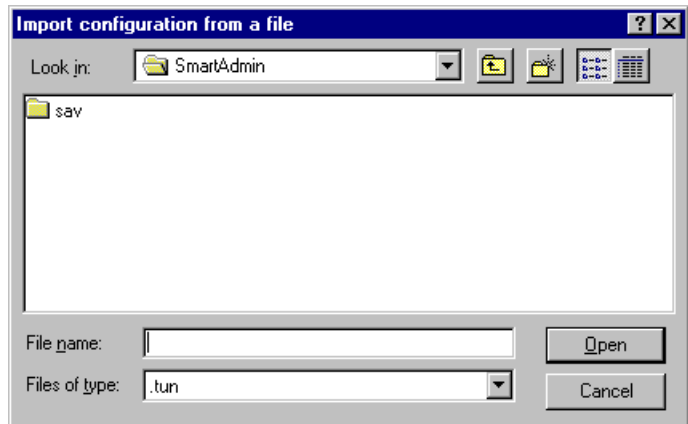
2. Fill in a File name. This is usually saved to a disk in the A: drive, and press **Enter**.

3. Transport this disk to the administrator on the remote end of the SmartGate IPSec tunnel. This can be accomplished by mail or the contents of the tunnel file can be sent by encrypted e-mail. This file then needs to be imported by the remote-end administrator. See Import below.

### Importing a Site-to-Site IPSec Tunnel

The Import button displays a Read from File dialog pointing to an appropriate directory (on the SmartAdmin client machine), showing all \*.tun files. Figure 11-16 is displayed.

**Figure 11-16**  
**Import Configuration from a File Window**



1. Select a file and click **open**.

This file is read and the tunnel name is compared to all existing tunnels. If a matching tunnel name is not found, it also searches for a match of network addresses in `extrn.nets`. If a match is found by name and address, the new tunnel replaces the old with no questions asked. If a match is found by network address or name, but not both, the user is first asked if the existing tunnel should be replaced, or if a new tunnel should be created (warning that the two tunnels have conflicting address spaces/names). Note that it is acceptable for two tunnels to have the same gateway addresses (to allow, for example, an “encrypted” and an “auth-only” tunnel between the same two sites), but it is not acceptable for two tunnels to have the same name.

### Editing an IPSec Site-to-Site Permission

The Edit IPSec Site-to-Site Permission Windows opens a dialog similar to the Add Permission. The Edit Window allows direct editing of the keys for the tunnel. However, the administrator will NOT be able to select encryption and authentication types

here—only to modify the keys and SPIs based on the current ESP/AH settings in the channel. Changes made with Manual Key Edit will take effect on the next.\* fields (not the current keys) and will only be saved to the file when the main edit dialog is completed (not when the menu key edit dialog is completed).

## Generating New Keys

New keys should be generated a various time intervals for security reasons.

1. Select a site from the list of IPSec tunnels on the IPSec Site-to-Site Configuration Tab.
2. Press **Export**.  
When this is pressed, an entire set of next.\* fields will be made, based on the setting of channel as well as keys and security parameter index's generated at random. All selected tunnels will have new keys generated.
3. Send the new exported tunnel description to the remote-site administrator (this information can now be sent through e-mail if it goes through the tunnel.)
4. The SmartGate administrator must press **Import** and load the new tunnel description.
5. Once the administrator's at both ends of the tunnel have the new keys, they should select the same site record and simultaneously press **Deploy New Keys**.

## Deploying New Keys

This button will replace the currently active keys with the keys that were generated the last time the **Generate New Keys** button was pushed. Separating the generation of new keys from their activation allows the administrator time to export the new keys and ship them over to the remote SmartGate via the existing secure tunnel.

## EXAMPLE

Following is an example of a `sites.ac1` file that contains the information for two tunnels:

1. A tunnel to an office in Seattle, which has a 192.168.2.0/24 network, and gateway at 1.2.3.4.
2. A tunnel to an office in Houston, which has 10.2.0.0/16 and 172.16.2.0/24 networks, and gateway at 4.3.2.1.

**NOTE:** A set of keys will also be generated at the start of the **Add** procedure, as well as any time the setting of Channel is changed, and any time the encryption parameters in the set channel are found to be different than they were last time any keys were generated.

The local network is 10.0.0.0/16. The Seattle office and Houston office are connected via the Central office—this is why the `intrn.nets` for Seattle's tunnel includes the Houston networks, and the `intrn.nets` for Houston's tunnel includes the Seattle network.

```
[Site]
name=Central-Seattle
intrn.nets=10.0.0.0/16,10.2.0.0/16,172.16.2.0/24
extrn.nets=192.168.2.0/24
channel=encrypted
remote_end=1.2.3.
tx.ipcomp=2          # IPCOMP_DEFLATE
tx.esp.spi=1a5f6edb
tx.esp.crypt_transform=3 # ESP_3DES
tx.esp.crypt_key=10f3c45ab3084c332304abdce040349657
tx.esp.auth_attribute=3 # AH_SHA
tx.esp.auth_key=c45ab3084c332304abdce04034965745890a89b0980d89c0e
rx.ipcomp=2          # IPCOMP_DEFLATE.
rx.esp.spi=24378da8
rx.esp.crypt_transform=3 # ESP_3DES
rx.esp.crypt_key=10f3c45ab3084c332304abdce040349657
rx.esp.auth_attribute=3 # AH_SHA
rx.esp.auth_key=c45ab3084c332304abdce04034965745890a89b0980d89c0e

# These are new keys for the tunnel, which have been
# exported to the remote end, but haven't
# yet been deployed.

tx.ipcomp=2          # IPCOMP_DEFLATE
tx.esp.spi=1a5f6f34
tx.esp.crypt_transform=3 # ESP_3DES
tx.esp.crypt_key=10f3c45ab3084c332304abdce040349657
tx.esp.auth_attribute=3 # AH_SHA
next.tx.esp.auth_key=c45ab3084c332304abdce04034965745890a89b0980d89c0e
rx.ipcomp=2          # IPCOMP_DEFLATE
rx.esp.spi=1a5f6f32
rx.esp.crypt_transform=3 # ESP_3DES
next.rx.esp.crypt_key=10f3c45ab3084c332304abdce040349657
rx.esp.auth_attribute=3 # AH_SHA
next.rx.esp.auth_key=c45ab3084c332304abdce04034965745890a89b0980d89c0e

[Site]
name=Central-Houston
intrn.nets=10.0.0.0/16,192.168.2.0/24
extrn.nets=10.2.0.0/16,172.16.2.0/24
channel=encrypted
remote_end=4.3.2.1
tx.ipcomp=2          # IPCOMP_DEFLATE
tx.esp.spi=1a5f4223
tx.esp.crypt_transform=3 # ESP_3DES
tx.esp.crypt_key=bf7890c78907890e7890a7809a7890e7890
tx.esp.auth_attribute=3 # AH_SHA
tx.esp.auth_key=fb89-c890a890bc890ef890d8c89b0890a562354a34c364de
rx.ipcomp=2          # IPCOMP_DEFLATE
rx.esp.spi=32cab974
rx.esp.crypt_transform=3 # ESP_3DES
rx.esp.crypt_key=17890bbf7890c7890d7890adc789d4563b2
rx.esp.auth_attribute=3 # AH_SHA
rx.esp.auth_key=5427890bc45da234512890-bcda46e3461289d89cb890a8cd
```





**3DES:** See [“Triple DES Encryption”](#)

**Access Code:** The secret code, similar to a PIN on an ATM card—required to unlock the authentication key stored on the user’s token each time the user accesses a secure service. This code, defined by the user during registration, must be at least four characters in length with a maximum of 16, and can be any combination of letters and numbers.

The length of time the system will ‘remember’ the Access Code is displayed in the **Remember code for: xxx minutes** field in the Access Code Dialog Box. The default limit is 10 minutes, however, the user may change the time limit by entering a new time (number of minutes—maximum of 999). SmartPass resets the limit every time Dynamic Configuration is performed.

**Access Control:** Allowing or denying connections through the use of access permissions.

**Access Permissions:** The associations between users and connections, as defined by a User ID, group name, service (TCP or Web), or destination. SmartGate access permissions can be either individual user permissions or group permissions.

**Administrator ID:** The identifier assigned to the SmartGate Server administrator when the administrator is added to the server. The NT Administrator ID is usually “Administrator” and the UNIX Administrator ID is usually “root.”

**Authentication:** The process of determining the identity of a user attempting to access a system.

**Authentication Key:** The key is a 32-character hexadecimal key assigned to a user during installation by the registration server administrator, consisting of the numbers 0 to 9 and letters A to F.

The SmartGate authentication system supports virtual smart cards and ISO-standard smart cards for both authentication and stored data. A user with a physical smart card must use a smart card reader connected to their PC. Virtual smart card information (VCAT token) may be stored on either the PC hard drive or a removable (floppy) disk.

The user’s SmartGate authentication key is stored on the smart card, whether physical or virtual. This information is shared with the SmartGate Server, where it is stored in the SmartGate Server’s user database.

**Authentication Header (AH)—IPSec:** Provides a packet-by-packet authentication of the “entire” packet, including the IP header. V-ONE supports MD5-HMAC, and SHA-HMAC.

**Authentication Token:** A portable device used for authenticating a user. Authentication tokens operate by challenge/response, time-based code sequences, or other techniques. This may include paper-based lists of one-time passwords.

**Authenticator:** The name assigned to a SmartGate Server through which users can access a particular service. This name can be up to 14 alphanumeric characters in length and it is recommended that it be a derivative of your SmartGate Server hostname.

**BSD:** Berkeley Software Distribution. A version of UNIX developed by the Computer Systems Research Group at the University of California at Berkeley. BSD enhancements include networking, virtual memory, job control, and large file names.

**BSDI:** Berkeley Software Design, Inc.

**Challenge/Response:** An authentication technique whereby a server sends an unpredictable challenge to the user. The user then computes a response using some form of authentication token.

**CHIPDRIVE external Smart Card Reader:** A device, developed by TOWITOKO electronics, used to read the information contained on a physical smart card. The CHIPDRIVE external card reader plugs directly into the serial port and does not need a battery.

**Client Port:** The TCP/UDP port number your application will use to gain access to the secured services on a TCP/IP network. This is used when you want to make connections via `localhost` rather than the shim and is configured by the SmartGate administrator in the access control list.

*For Example:*

21	Used for FTP
2023	Used for <a href="#">Telnet</a>
25	Used for <a href="#">SMTP</a> (e.g., sendmail services)
110	Used to retrieve messages using POP3 mail protocol

**Client/Server:** Computer technology that separates computers and their users into two categories: clients or servers. When you request information from a computer, you are a client. The computer that provides the information is the server. A server both stores information and makes it available to any authorized client who requests the information.

**Customer ID:** The customer's identification number. The Customer ID along with the Serial Number, which also identifies the customer, are used to generate a License Key and certificate. You will have received your Customer ID by e-mail from V-ONE upon registration of the software.

**DES Encryption:** Data Encryption Standard. A U.S. government-approved method of encryption that currently uses a 56-bit key.

**Digital Certificate:** A digital certificate is an electronic “credit card” that establishes your credentials when doing business or other transactions on the Web. It is issued by a certification authority (CA). It contains your name, a serial number, expiration dates, a copy of the certificate holder’s public key (used for encrypting and decrypting messages and digital signatures), and the digital signature of the certificate-issuing authority so that a recipient can verify that the certificate is real. Some digital certificates conform to a standard, X.509. Digital certificates can be kept in registries so that authenticated users can look up other users’ public keys.

**Disable SmartGate User:** Temporarily disables a User ID from any access through the SmartGate Server. Disabled users do not reflect in the license use.

**DNS:** See “*Domain Name Service*.”

**Domain Name:** Identifies a ‘location’ on the Internet (e.g., `v-one.com`) that has been registered with the Internet Network Information Center (InterNIC). Currently the domain name is limited to 47 characters. Through the use of aliases, however, it is possible to accommodate longer names. You may contact V-ONE Corporation for support in this connection.

**Domain Name Service (DNS):** The distributed database that maps Internet domain names to IP addresses or IP addresses to Internet domain names, in addition to containing other information.

**Enable SmartGate User:** Reinstates a disabled User ID.

**Encapsulating Security Payload (ESP)—IPSec:** ESP provides the encryption part of IPSec, as well as optionally calculating a packet-by-packet authentication hash of the “payload” of each packet (but not the IP header).

**End User:** Those users who connect to a SmartGate Server using the SmartPass client software on their personal computers.

**Entrust Authentication:** The Entrust authentication method allows SmartPass users to use an Entrust soft token as an alternative authentication method. Entrust provides digital certificates to create an on-line identification and security system for the Internet. Both SmartPass and the SmartGate/Entrust Server must obtain their credentials from the Entrust CA Server, enabling both sides to validate the other party during the authentication process.

**Entrust Certification Authority (CA) Server:** The Entrust Server representing the organization or group of people who are responsible for setting security policies regarding the protection of sensitive and valuable data and assigning secure electronic identities in the form of certificates. These people are referred to collectively as the Certification Authority (CA).

**Entrust/Netrust Authorization code:** The Authorization code issued by the Entrust or Netrust CA Server.

**Entrust/Netrust Directory:** The directory where your Entrust or Netrust files (specifically `entrust.ini` and, when using UNIX, the run-time library files) are installed. The `entrust.ini` file contains the location of the Entrust or Netrust CA Server and Manager and is used by both the SmartGate Server and SmartPass. The `entrust.ini` file is obtained from Entrust or Netrust—not V-ONE—It is necessary for the operation of the software.

**Entrust/Netrust Reference Number:** The Reference number issued by the Entrust or Netrust CA Server.

**FIPS 140-1:** Compliance with FIPS 140-1 government coding standards.

**FIPS Token:** A software emulation of a hardware authentication token that is in compliance with the FIPS 140-1 coding standards. It stores your private information (authentication key) in an encrypted file system, either on a floppy disk or on your hard drive.

**Firewall:** A system or combination of systems that enforces network access policies between two or more networks.

**Firewall Host Name (or IP Address):** This is either a valid DNS name or an actual IP Address (e.g., `206.133.19.26`) used to connect to a SmartGate Server.

**Firewall Port:** This is the TCP/IP port number the SmartGate Server will be listening on for secure connection requests by SmartPass (e.g., 2023, 2021). This port will be assigned by the SmartGate Server administrator.

**Firewall-to-Firewall Encryption:** All traffic from one firewall to another over the Internet is automatically encrypted.

**Format Smart Card:** This function erases the current secret format code from a smart card, and returns it to the default secret code. The authentication key is **NOT** removed during the formatting process.

**Format Code:** An 4- to 16-character code which allows the user to format a smart card.

**FQDN:** Fully Qualified [Domain Name](#). This is a hostname which will include the hostname and domain name. For example, the machine “test” within the domain “v-one.com”—it’s FQDN would be “test.v-one.com”.

**FTP:** FTP (File Transfer Protocol) is a way of moving one or more files from one computer to another on the same network. It is especially useful when the files are too large to fit on a floppy disk and when moving files between different operating systems.

**G&D STARCOS Smart Card:** A microprocessor-based physical smart card manufactured by Giesecke & Devrient GmbH (G&D).

**Gemplus MCOS Smart Card:** A microprocessor-based physical smart card manufactured by Gemplus.

**Integrity:** The assurance that any data has not been altered in transmission.

**IP:** Internet Protocol.

**IP Payload Compression (IPCOMP)—IPSec:** A method of compressing the data in the payload of an IP packet so that it will take up less bits on the wire. V-ONE supports the DEFLATE protocol.

**IPSec:** Internet Protocol Security (IPSec). A suite of protocols used for secure private communications over the Internet. The proposed suite of IPSec protocols would create a standard platform for securing IP connections on private networks. These protocols basically deal with authentication, encryption, and key management.

**ISA:** Industry Standard Architecture. An expansion bus commonly used in PCs, it accepts plug-in boards currently being superseded by “PCI” boards.

**Key:** See “[Authentication Key](#)”

**License Key:** The License Key is a file (`vone.lic`) generated by V-ONE from a combination of the Customer ID, Serial Number, the number of SmartGate users, the expiration date of the SmartGate software, and any additional features encoded into a single key, without which your SmartGate Server software will not run. Your Serial Number and Customer ID are necessary in order to obtain your License Key.

**Linux:** Linux is a UNIX operating system clone which runs on a variety of platforms, especially personal computers with Intel 80386 or better processors. It supports a wide range of software, from TeX, to the X Window System, to the GNU C/C++ compiler, to TCP/IP. It is a versatile, bona fide implementation of UNIX, freely distributed under the terms of the GNU General Public License.

**Logging:** The process of storing information about events that occurred on the firewall or network.

**Log Retention:** How long audit logs are retained and maintained.

**Log Processing:** How audit logs are processed, searched for key events, or summarized.

**MCOS:** See “[Gemplus MCOS smart card](#)”

**Mutual Authentication:** Bidirectional authentication where the client is required to authenticate to the server, and the server is required to authenticate to the client.

**Netrust Anonymous Registration:** Anonymous registration allows a Netrust end user to log in and register to the SmartGate/Netrust Server without performing OLR. When a user is registered anonymously, they are identified in the SmartGate user database by an ID derived from their Netrust smart card’s serial number, which displays as a random string of numbers.

**Netrust Authentication:** The Netrust authentication method allows SmartPass users to use a Netrust ready smart card and smart card reader instead of other V-ONE tokens, such as a VCAT token. Both SmartPass and the SmartGate/Netrust Server obtain their credentials from the Netrust Certificate Authority (CA) Server. Both sides will validate the other party during the authentication process. See *Using SmartGate With Netrust Authentication* for complete information on Netrust.

**Netrust Certification Authority (CA) Server:** The Netrust Server representing the organization or group of people who are responsible for setting security policies regarding the protection of sensitive and valuable data and assigning secure electronic identities in the form of certificates. These people are referred to collectively as the Certification Authority (CA).

**Network Adaptor Card:** A card that connects a terminal device to the network.

**Non-repudiation:** Ensuring that the original message has not been altered since it was sent.

**On-Line Registration (OLR):** All SmartPass end users must register before they can access the system. OLR provides a means by which users can register at their computer, immediately after installing the SmartPass software.

**Password:** A sequence of characters which, when combined with a user name, limits a log-on only to the authorized user.

**PCAT Parallel Smart Card Reader:** A device used to read the information contained on a physical smart card.

**Physical Smart Card:** Credit card-sized device implanted with integrated circuit chips used for a variety of applications, such as financial debt/credit transactions and computer security.  
*See also “[Smart Card Technology](#)”*

**PKI:** The public key infrastructure provides for digital certificates that can identify individuals or organizations and directory services that can store and, when necessary, revoke them.

**Privacy:** The authorized distribution of information (who has a right to know what).

**Private Key Cryptography:** An encryption method which requires both parties of a digital transmission to know the same key for encryption and decryption.

**Proxy:** A software agent that acts on behalf of a user. Typical proxies accept a connection from a user, make a decision as to whether or not the user or client IP address is permitted to use the proxy, may perform additional authentication, and then complete a connection on behalf of the user to a remote destination.

**Public Key Cryptography:** An encryption method that allows secure communication between two parties who have not transmitted a private key in advance. Each party transmits a public key used to encode messages to itself. Public key messages can only be decoded with a private key which is never transmitted.

**RADIUS Authentication:** RADIUS authentication is an open-standard (RFC 2138) authentication protocol transported over UDP, not TCP.

**Remote Authentication Dial-In User Service (RADIUS):** An authentication method using a username/password combination with MD5 hashing of password for increased security. Described in RFC 2138.

**Remote Host Name:** Either a valid DNS name or an actual IP Address (e.g., 206.133.19.26) that identifies the host on which the secured service is provided. A SmartGate Server will connect to this host after SmartPass completes an authenticated session connection. You will be given the correct value to use for each designated service by your SmartGate Server administrator.

**Remote Host Service Port:** This is the port number the designated remote service host will be listening on to accept connections for service from a SmartGate Server after any user is successfully authenticated for service.

**RSA SecurID Authentication:** A dual-factor authentication method using the RSA SecurID token and ACE/Server authentication products developed by RSA Security, Inc. SmartGate supports all types of RSA SecurID authentication tokens, including the standard card/key fob, PINPAD card, and SoftID card.

**Rule Set:** The group of instructions that determine distribution of system privileges to users.

**SCSI (Small Computer System Interface):** Pronounced “scuzzy,” SCSI is a hardware interface that allows for the connection of up to seven peripheral devices (hard disk, CD-ROM, scanner, etc.) to a single expansion board in a computer.

**Security:** Protection or defense against unauthorized access to data. The four attributes of security are: [Authentication](#), [Integrity](#), [Privacy](#), and [Non-repudiation](#).

**Serial Number:** The numeric string identifying the customer. The Serial Number along with the Customer ID, which also identifies the customer, are used to generate a License Key and certificate. You will have received your Serial Number by e-mail from V-ONE upon registration of the software.

**Server:** See “[Client/Server](#)” or “[SmartGate Server](#)”

**Single Port Proxy:** The SmartGate Single Port Proxy provides the various SmartGate services with a single-port presence on the perimeter of a network. Therefore, all SmartPass-to-SmartGate connectivity will pass through the Single Port Proxy and be forwarded to the correct destination SmartGate service. The Single Port Proxy is defaulted at 3845 and can be changed with the “deloyability” option.

**Smart Card:** See “[Physical Smart Card](#)” or “[Virtual Smart Card](#)”

**Smart Card Reader:** A device used to read the information contained on a physical smart card. At present, SmartPass supports V-ONE’s PCAT parallel smart card reader, Fischer’s Smarty reader, and TOWITOKO electronics’ CHIPDRIVE external card reader.

**Smart Card Technology:** Credit card-sized device implanted with integrated circuit chips used for a variety of applications, such as financial debt/credit transactions and computer security, a device to read the information contained on the card, and software controls.

**SmartAdmin:** SmartGate’s stand-alone administrative software, which is used to manage user information and access permissions, including administrative rights. Using SmartAdmin, a SmartGate administrator may also configure the SmartGate Server and its Single Port proxy mapping rules. SmartAdmin runs remotely on a Windows 95 or 98 and either remotely or locally on a Windows NT Server.

**SmartGate Group:** Users grouped by organization, such as department (e.g., accounting or sales), location (e.g., Chicago), or artificial community, such as a project being managed as an independent set of services (e.g., ProjectX). A SmartPass end user belongs directly to one SmartGate group. However, in addition to the group “all,” the user can be given the permissions of as many groups as desired. The group’s name can be up to 23 characters in length and cannot include spaces or special characters.

**SmartGate Hostname or IP Address:** Either a valid DNS name or an actual IP address (e.g., 206.133.19.26) used to connect to a SmartGate Server.

**SmartGate Port:** This is the TCP/IP port number the SmartGate Server will be listening on for secure connection requests by SmartPass (i.e., 2023, 2021).

**SmartGate Server:** The machine running the SmartGate Server software. Logically, this is the machine between the user’s personal computer and the ultimate destination for the application being secured by SmartGate.

**SmartGate Server Administrator:** The person responsible for maintaining the SmartGate Server and user authentication database on the SmartGate Server.

**SmartGate Web Group:** A group of SmartGate users or groups requiring access to the same remote host services.

**NOTE:** This applies only to Web services which may be accessed through the SmartGate Server.

**SmartPass:** The client software that runs on the end user’s personal computer and is used to connect to the SmartGate Server.

**Smarty Reader:** A device, developed by Fischer International, used to read the information contained on a physical smart card. The Smarty reader simulates a 3.5-inch computer disk. The physical smart card is inserted into the Smarty reader which, in turn, is inserted into the computer’s floppy drive.

**SMTP:** Simple Mail Transfer Protocol.

**Soft Token:** See “[FIPS Token](#)” or “[VCAT Token](#)”

**STARCOS:** See “[G&D STARCOS Smart Card](#)”

**TCP/IP (Transmission Control Protocol/Internet Protocol):** This is the suite of protocols that defines the Internet.

**Telnet:** An Internet protocol and program that enables you to connect your PC as a remote workstation to a host computer, and to use that computer as if you were logged on locally.

**Triple DES Encryption (3DES):** A method of encryption that uses a 168-bit key. There are four separate situations in which 3DES encryption is an option; SmartGate protocol packets used to manage client/server secure communications, proxy data packets used to convey end user data between client and Server, and internal server communication between the Authentication Server and the proxy. 3DES is also an option when configuring specific IPsec channel types.



**Uniform Resource Locator (URL):** The server and path information used to specify the location of a document. Formatted as follows:

`scheme://host-domain[:port]/path/filename`

The maximum length allowed for a URL in SmartPass is 256 characters.

**User ID:** The identification associated with the authentication key which is either generated during OLR or assigned by the SmartGate Server administrator when the user is added to the SmartGate Server database. A User ID may be up to 30 characters in length and cannot include spaces or special characters. However, your User ID defaults to the authentication token manufacturer's parameters. For example, the MCOS and STARCOS physical smart cards have a limit of 15 characters for the User ID. Each User ID must be unique on the SmartGate Server where it resides.

**User Name:** See “*User ID*”

**VCAT Token:** A software emulation of a hardware [authentication token](#). It stores your private information ([authentication key](#)) in an encrypted file system, either on a floppy disk or on your hard drive.

**Virtual Private Network (VPN):** A private network created over a public network (e.g., the Internet) by using encryption, where exclusive client and host communications can occur.

**Virtual Smart Card:** See “[FIPS Token](#)” or “*VCAT Token*”

**World Wide Web:** Generally used to refer to the whole constellation of Internet resources that can be accessed using Gopher, FTP, HTTP, Telnet, USENET, WAIS and other tools. Also, the universe of hypertext (HTTP) servers that allow text, graphics, sound files, etc., to be mixed and accessed.



# Appendix A

## SmartGate Server Files

This appendix contains detailed descriptions of the SmartGate Server files that you may create or customize, organized alphabetically by file type. Following the file descriptions is a section containing explanations of each of the configuration options available for the SmartGate Server configuration file, `sgconf.ini`.

### SmartGate Server Files Detailed Descriptions

#### Access Control Lists

The following section contains descriptions of files for [access control](#). Only the Authentication Server needs these files.

##### **ipsec.acl**

SmartGate Server Access Control List

**Used By:** The Authentication Server (`sgasrv`).

**Purpose:** Provides access control for IPSEC-secured IP services including E-mail, FTP, Oracle, Telnet Server, **rlogin** Proxy support, etc.

**Location:** Installed in the SmartGate Server's root directory on a Microsoft Windows NT Server.

**NOTE:** IPSec is only available with SmartGate 4.0 and later running on a Microsoft Windows NT server.

**Structure:** This file uses sections for users and groups. All group names must be preceded by a tilde (~). Comments must be preceded by a semicolon (;) or a pound sign (#). The format of a section header is:

[~group]

or:

[userid]

*group* is the group for which access is being defined, or ~all if access is being defined for all users. Group names must be preceded by a tilde (~).

*userid* is the individual user's User ID for which access is being defined.

Each section can contain three types of access permissions:

1. Path permission
2. DNS proxy
3. Include group of permissions

**Path permissions:** Each path permission is in a single line, starting with the string `policy:` and containing multiple *key=value* pairs separated by spaces. The *key=value* pairs that are valid in a `policy` line are:

#### **channel**

Type: ASCII string  
Meaning: name of a channel type to be used for this policy (from `chantype.ini`)  
Default: "encrypted" - use the default channel type for this SmartGate Server

#### **action**

Type: ASCII String  
Meaning: PASS or REJECT  
Default: PASS

#### **priority**

Type: ASCII-decimal representation of unsigned 32bit number  
Meaning: priority of this rule relative to other rules— from lowest to highest  
Default: Medium

**NOTE:** Case is not significant in the `domain_name` string.

### **ipproto**

Type: ASCII-decimal 0-255, or string constant  
Meaning: IP protocol number (e.g., “6” for TCP, “17” for UDP)  
Default: 0 (“all IP protocols”)

### **intrn.addr.pub**

Type: <dotted quad>  
Meaning: IP address or hostname of internal host or network allowed by this rule  
Default: 0 - any address (assuming .mask is also 0)

### **intrn.mask**

Type: <dotted quad>  
Meaning: netmask to apply to above addresses when matching and when computing a NAT address  
Default: 0.0.0.0

### **intrn.port**

Type: ASCII-decimal 0-65535  
Meaning: TCP or UDP port to match rule  
Default: 0 - match all ports

**DNS Proxies:** Each DNS Proxy is a single line, starting with the string `dnsproxy:` and containing multiple *key=value* pairs separated by spaces. The *key=value* pairs that are valid in a `dnsproxy` line are:

### **domain\_name**

Type: arbitrary length string  
Meaning: domain name whose DNS requests should be proxied to this server e.g., “\* .v-one .com”  
Default: no default

### **dns\_server**

Type: <dotted quad>  
Meaning: address where requests within `domain_name` should be redirected  
Default: no default

**“Group Include” permissions:** A section may include another group section by placing that group’s name (including the leading ~), preceded by `group:` on a line within the section. For example, to include the “engineer” group in the user “joe” section:

```
[joe]
...
group: ~engineer
...
```

The effect of this is that all access permissions for the group ~engineer (and all access permissions for any group included by ~engineer) will now be applied to user joe. Includes can be nested to an arbitrary depth. However any recurrence of a group name within the hierarchy will cut off any other included groups or access permissions.

User sections cannot be included in other sections—only a group section can be included.

Access permissions in the group ~all are automatically assigned to all end users.

**Customization:** You can edit `ipsec.acl` using SmartAdmin, either remotely or locally.

#### Example:

```
[joe]
policy: intrn.addr.pub=10.0.20.11 intrn.port=23
ipproto=6
policy: intrn.addr.pub=10.0.21.0
intrn.mask=255.255.255.0
group: ~engineer
[~engineer]
policy: intrn.addr.pub=137.175.2.0
intrn.mask=255.255.255.0
[~all]
policy: intrn.addr.pub=192.168.2.1 ipproto=tcp
dnsproxy: domain_name=*v-one.com
dns_server=10.0.0.11
.
.
```

## sites.acf File

Site-to-Site tunnel configuration file

**Used by:** The IPSec Service

**Purpose:** Provides configuration information for the local ends of the IPSec tunnels for each listed remote site.

**Location:** Installed in the SmartGate Server's root directory\data directory.

**Structure:** `sites.acf` is structured like a Microsoft Windows's .ini file, with sections separated by lines containing the word **Site** in square brackets. Each section contains all the configuration information needed for a particular remote site's tunnel. Each field is on a separate line. A field's name and value will be separated by an =, with no intervening spaces, e.g., *name=value*. NO SPACES are allowed in any name or value. Some fields may be extremely long, in which case the \ character can be used to continue the field on the following line. No line in the file should be more than 255 characters long—if it is longer than this, it should be split into multiple lines

### Individual Fields

Following are the fields allowed in each section, and their descriptions.

1. Policy and Scheme information are combined since there is only one policy per site. There is a 1:1 relationship between policies and schemes.
2. Only policy addresses can be configured—no setting of port, protocol, or direction is allowed.
3. All keying (scheme-related) information has a **fieldname** field as well as a **next.fieldname** field. The “next” is in place so that the master end can generate new keys for the tunnel without immediately putting them into service, thus allowing the current keys to continue working until the new ones can be exported and transferred to the remote end.

## General

name

- Type: ASCII String
- Meaning: Name of this site-to-site connection. NO SPACES ALLOWED!
- Default: None - Required Field

## Policy-related

intrn.nets

- Type: ASCII string containing a comma-separated list of:
1. dotted quads (eg "1.2.3.4")
  2. dotted quads with netmask specifier (e.g., 192.168.2.0/24)
- Meaning: List of network addresses on the local private network that will be accessible to the remote network via this tunnel.
- Default: None - required field

extrn.nets

- Type: ASCII string containing a comma-separated list of:
1. dotted quads (eg "1.2.3.4")
  2. dotted quads with netmask specifier (e.g., 192.168.2.0/24)
- Meaning: List of network addresses on the remote private network that will be accessible to the local network via this tunnel.
- Default: None - required field

## Scheme-related:

enable

- Type: ASCII numeral 1 or 0
- Meaning: 1 if this tunnel is currently enabled, else 0.
- Default: 1

Date

- Type: ASCII string in the format yyyy/mm/dd-HH:MM:SS (note the - separating date from time rather than space)
- Meaning: Time and date that the currently active keys were generated
- Default: None

**NOTE:** Spaces are \*not\* allowed within or between fields.

**NOTE:** Do not use a domain name for `intern.nets` or `extern.nets`, IP addresses are accepted only.



**NOTE:** Do not use a domain name for `local_end`, IP addresses are accepted only.

**NOTE:** Do not use a domain name for `remote_end`, IP addresses are accepted only.

**NOTE:** Each of the following fields can be repeated with a preceding “next.”

**NOTE:** To make the configuration process simpler, SmartGate will always return the SPI(s) the client sent to it.

Channel	
Type:	ASCII string
Meaning:	Name of a channel type to be used for this policy (from <code>chantype.ini</code> )
Default:	“Default.” Use the default channel type for this SmartGate server
gateway	
Type:	ASCII string
Meaning:	Which IPsec server to use as the server-side gateway for this policy.
Default:	“Default.” Use the default IPsec gateway for this SmartGate Server (usually <code>localhost</code> ).
local_end	
Type:	<dotted quad>
Meaning:	IP Address of client end of tunnel (optional)
Default:	Use the setting of “gateway.” Note that <code>local_end</code> generally should not be used at all. Use gateway instead, or set the <code>ipsec_server</code> address in <code>sgconf.ini</code> .
remote_end	
Type:	<dotted quad>
Meaning:	Address of remote end of tunnel
Default:	None - required field
The following items are repeated for each choice in [   ]’s	
[tx rx].ipcomp	
Type:	Decimal number (based on <code>IPCOMP_* rfc2407</code> )
Meaning:	IP compression type to use
Default:	0 - none
[tx rx].[esp ah].spi	
Type:	ASCII-hexadecimal 32 bit
Meaning:	SPI for this direction and transform
Default:	None - required if an SA of the given type is used.
[tx rx].esp.crypt_transform	
Type:	Decimal number (based on <code>ESP_* rfc2407</code> )
Meaning:	What type of encryption to use
Default:	0 - none

`[tx|rx].esp.crypt_key`

Type: Arbitrary length ASCII-hex string

Meaning: Key info for encryption

Default: Empty

`[tx|rx].ah.auth_transform`

Type: Decimal number (based on AH\_\* RFC 2407)

Meaning: What type of AH authentication to use

Default: 0 - none

`[tx|rx].[esp|ah].auth_attribute`

Type: Decimal number (based on AUTH\_ALG\_\* RFC 2407)

Meaning: What type of ESP/AH auth to use

Default: 0 - none

Since ISAKMP defines both the `ah.auth_transform` and `ah.auth_attribute`, we do the same. In order to do RFC 1826-compliant AH, you must use one of the following pairs:

AH\_MD5 / AUTH\_ALG\_KPDK

AH\_SHA / AUTH\_ALG\_KPDK

`[tx|rx].[esp|ah].auth_key`

Type: Arbitrary length ASCII-hex string

Meaning: Key info for ESP/AH-auth

Default: None

### **[next.]\***

Type: ASCII numeral 1 or 0

Meaning: 1 if next.\* fields are valid, else 0

Default: 0

### **The Meaning of "next."**

It may be desirable to modify a site-to-site tunnel's parameters but delay the enactment of those changes until the remote gateway has a chance to obtain the new parameters. In particular, an administrator may wish to generate new keys for the tunnel without any long interruption in service. In this case, the administrator should generate new keys and other encryption parameters, and store them in `sites.acl` with the string "next." prepended to each field name. The key export operation will then export this new key information, it will be sent to the remote, and the administrator at the remote end will import it. Once both ends are ready (with the new key

information in the “next.\*” fields), they will both press the “activate new keys” buttons for their respective tunnels at the same time, thus switching both ends to the new keys. At this time, the next.\* keys will be moved in place of the “non-next” keys in `sites.ac1` at both ends.

## EXAMPLE

Following is an example of a `sites.ac1` file that contains the information for two tunnels:

1. A tunnel to an office in Seattle, which has a 192.168.2.0/24 network, and gateway at 1.2.3.4.
2. A tunnel to an office in Houston, which has 10.2.0.0/16 and 172.16.2.0/24 networks, and gateway at 4.3.2.1.

The local network is 10.0.0.0/16. The Seattle office and Houston office are connected via the Central office—this is why the `intrn.nets` for Seattle’s tunnel includes the Houston networks, and the `intrn.nets` for Houston’s tunnel includes the Seattle network.

```
[Site]
name=Central-Seattle
intrn.nets=10.0.0.0/16,10.2.0.0/16,172.16.2.0/24
extrn.nets=192.168.2.0/24
channel=encrypted
remote_end=1.2.3.
tx.ipcomp=2          # IPCOMP_DEFLATE
tx.esp.spi=1a5f6edb
tx.esp.crypt_transform=3 # ESP_3DES
tx.esp.crypt_key=10f3c45ab3084c332304abdce040349657
tx.esp.auth_attribute=3 # AH_SHA
tx.esp.auth_key=c45ab3084c332304abdce04034965745890a89b0980d89c0e
rx.ipcomp=2          # IPCOMP_DEFLATE.
rx.esp.spi=24378da8
rx.esp.crypt_transform=3 # ESP_3DES
rx.esp.crypt_key=10f3c45ab3084c332304abdce040349657
rx.esp.auth_attribute=3 # AH_SHA
rx.esp.auth_key=c45ab3084c332304abdce04034965745890a89b0980d89c0e
```

```
# These are new keys for the tunnel, which have been
# exported to the remote end, but haven't
# yet been deployed.
```

```
tx.ipcomp=2          # IPCOMP_DEFLATE
tx.esp.spi=1a5f6f34
tx.esp.crypt_transform=3 # ESP_3DES
tx.esp.crypt_key=10f3c45ab3084c332304abdce040349657
tx.esp.auth_attribute=3 # AH_SHA
next.tx.esp.auth_key=c45ab3084c332304abdce04034965745890a89b0980d89c0e
rx.ipcomp=2          # IPCOMP_DEFLATE
rx.esp.spi=1a5f6f32
rx.esp.crypt_transform=3 # ESP_3DES
next.rx.esp.crypt_key=10f3c45ab3084c332304abdce040349657
rx.esp.auth_attribute=3 # AH_SHA
next.rx.esp.auth_key=c45ab3084c332304abdce04034965745890a89b0980d89c0e
```

```
[Site]
name=Central-Houston
intrn.nets=10.0.0.0/16,192.168.2.0/24
extrn.nets=10.2.0.0/16,172.16.2.0/24
channel=encrypted
remote_end=4.3.2.1
tx.ipcomp=2          # IPCOMP_DEFLATE
tx.esp.spi=1a5f4223
tx.esp.crypt_transform=3 # ESP_3DES
tx.esp.crypt_key=bf7890c78907890e7890a7809a7890e7890
tx.esp.auth_attribute=3 # AH_SHA
tx.esp.auth_key=fb89-c890a890bc890ef890d8c89b0890a562354a34c364de
rx.ipcomp=2          # IPCOMP_DEFLATE
rx.esp.spi=32cab974
rx.esp.crypt_transform=3 # ESP_3DES
rx.esp.crypt_key=17890bbf7890c7890d7890adc789d4563b2
rx.esp.auth_attribute=3 # AH_SHA
rx.esp.auth_key=5427890bc45da234512890-bcda46e3461289d89cb890a8cd
```

## sgate.acl

SmartGate Server Access Control List

**Used By:** The Authentication Server (sgasrv).

**Purpose:** Provides access control for secured TCP services including E-mail, FTP, Oracle, Telnet Server, **rlogin** Proxy support, etc.

**Location:** Installed in the SmartGate Server's root directory/etc on a UNIX-based server and the SmartGate Server's root directory\data directory on a Microsoft Windows NT.

**Structure:** This file uses sections for users and groups. All group names must be preceded by a tilde (~). Comments must be preceded by a semicolon (;) or a pound sign (#). The format is:

**NOTE:** Wildcarded IP addresses, DNS names, and destination ports are valid in `sgate.acl`. For more information refer to Appendix C, "ACL Wildcarding."

[~*name*]

*host port [lport] [rport]*

*name* is the group or the individual user for which access is being defined, or ~all if access is being defined for all users. Group names must be preceded by a tilde (~).

*host* is either the hostname or IP address of the Destination Server to which this user, users in this group, or all users may have access.

*port* is the port number of the Destination Server to which the user, users in the group, or all users have access.

*lport* is the port number that SmartPass will listen on for this secure path.

*rport* is the port number that the SmartGate Server will listen on for this secure path.

**Customization:** You can edit `sgate.acl` using SmartAdmin, either remotely or locally (on a Windows NT SmartGate Server). You can also modify it locally by running the `sgateacl` program from the root directory at the server console or by running `./setup` from the `/bin` directory on a UNIX SmartGate Server.

**NOTE:** To create a valid new group, you must use SmartAdmin, UNIX setup script, or command line.

**NOTE:** In order to administer remotely, your User ID must be added as a SmartGate administrator to `adm-gw.acl`.

### Example:

```
;All users may access this server.
[~all]
abc 23
;The Engineering Group may access these servers.
[~eng]
eng1 23
engmail 25
engmail 110
;Management may access all company network servers
[~management]
107.0.0.* 23
;John Smith may access this server.
[jsmith]
uxdev2 23
.
.
```

### sgate.dny

SmartGate Server Access Control Denial List

**Used By:** The Authentication Server (sgasrv).

**Purpose:** Denies access to specified TCP services including E-mail, FTP, Oracle, Telnet Server, *rlogin* Proxy support, etc.

**Location:** Installed in the SmartGate Server's root directory/etc on a UNIX-based server and SmartGate Server's root directory\data on Windows NT.

**Structure:** This file's structure is identical to the `sgate.acl` file.

**Customization:** You can edit `sgate.dny` using a text editor from the server console.

### Example:

```
;No users may access these servers.
[~all]
abc 23
10.0.4.382 25
;The Word_Processing group may not access these servers.
[~word_processing]
10.0.0.385 23
;John Smith may not access this server.
[John28924]
205.0.0.* 23
```

**NOTE:** Single users must not be preceded by a tilde (~).

## sweb.acl

Web Server Access Control List

**Used By:** The Authentication Server (sgasrv).

**Purpose:** Provides access control to your Web server.

**Location:** Installed in the SmartGate Server's root directory/etc on a UNIX-based server and SmartGate Server's root directory\data on Windows NT.

**Structure:** This file uses sections for users and groups. All group names must be preceded by a tilde (~). Comments must be preceded by a semicolon (;) or a pound sign (#). The format is:

[~name]  
url [rport]

*name* is the Web group for which access is being defined, or "all" if access is being defined for all users. Group names must be preceded by a tilde (~).

*url* is the Uniform Resource Locator (URL) for the destination to which this user, users in this group, or all users have access. As with any URL, the port number may be included. For example:

/www.ooo.com:80/

If the port number is not included in the URL, the default port number (80) is used.

*rport* is the port number that the SmartGate Server will listen on for this secure path.

**Customization:** You can edit sweb.acl using SmartAdmin, either remotely or locally (on a Windows NT SmartGate Server). You can also modify it locally by running the swebacl program from the root directory at the server console or by running ./setup from the /bin directory on a UNIX SmartGate Server.

**NOTE:** Wildcarded IP addresses, DNS names, and destination ports are valid in sweb.acl. For more information refer to Appendix C, "ACL Wildcarding."

**NOTE:** The maximum length allowed for a URL is 255 characters.

**NOTE:** There must be a "/" at the beginning of the URL and at the end of the destination (host domain and optional port).

**NOTE:** In order to administer remotely, your User ID must be added as a SmartGate administrator to adm-gw.acl.

**NOTE:** To create a valid new group, you must use SmartAdmin, UNIX setup script, or command line.

### Example:

```
;All users may access this server.  
[~all]  
/www.v-one.com/pub/  
;The Engineering Group may access these servers.  
[~engineering]  
/www.v-one.com/eng  
;John Smith may access this server.  
[jsmith]  
/www.v-one.com/eng/jsmith  
;etc.....  
;Management may access all company servers on port 80.  
[~management]  
/*.v-one.com:80/
```

**NOTE:** Single users must not be preceded by a tilde (~).

### sweb.dny

Web Server Access Control Denial List

**Used By:** The Authentication Server (sgasrv).

**Purpose:** Denies access to specified Web servers.

**Location:** Installed in the SmartGate Server's root directory/etc on a UNIX-based server and SmartGate Server's root directory\data on Windows NT.

**Structure:** This file's structure is identical to the sweb.acl file.

**Customization:** You can edit sweb.dny using a text editor from the server console.

### Example:

```
;No users may access these Web servers.  
[~all]  
/www.riskywebsite.com:80/  
/www.competitorsite.com:80/  
;The Word_Processing Group may not access these servers.  
[~word_processing]  
/www.v-one.com/finance  
;John Smith may not access this server.  
[John28924]  
/www.v-one.com/eng/projectX
```



**NOTE:** All programs that access the user database (UID Server, Dynamic Configuration Server, etc.) must reside on the same computer where the user database is stored.

**NOTE:** In order to administer remotely, your User ID must be added as a SmartGate administrator to `adm-gw.acl`.

**WARNING!** It is highly recommended that a backup be made of the user database before using the `dbrw` application.

**WARNING!** Using the `dbrw` application for reasons other than the stated purpose, and not following these instructions exactly could cause corruption of your user database.

**WARNING!** If you are transferring the text file using FTP, it must be in ASCII format (type `asc`).

## Database Files

The following section contains descriptions of database files.

### **sgusrdb**

User Database

**Used By:** The user database is accessed by `sgadm` for updating the Authentication Server (`sgasrv`).

**Purpose:** The user database stores the SmartGate User ID, enabled/disabled status, user's long name, group to which the user is assigned, and authentication key for each user.

**Location:** The user database (`sgusrdb`) is installed by default in SmartGate Server's root directory\data on a Windows NT or in the SmartGate Server's root directory/etc on a UNIX-based server.

**Structure:** N/A

**Customization:** The user database is not a text file. It can be managed using SmartAdmin, either remotely or locally (on a Windows NT SmartGate Server). You can also modify it locally by running the `sgadm` program from the root directory at the server console.

**Example:** N/A

### **dbrw**

User Database Read/Write Application

**Used By:** The SmartGate or network/system administrator.

**Purpose:** Allows the administrator to move the SmartGate user database (`sgusrdb`) from one platform to another (for example, from BSD/OS to Solaris). The `dbrw` application can be used on all operating systems to read/write the user database file into a text file.

**Location:** The `dbrw` application is installed directly into the SmartGate Server's root directory on a Windows NT or in the /SmartGate Server's root directory/bin on a UNIX-based server.

**Structure:** The `dbrw` application allows for two arguments.

1. **`dbrw -w filename`**

The user database is written to a specified file.

## 2. **dbrw -r filename**

Either a new SmartGate user database is created from the formatted text file or the text file is read into an existing user database.

If the text file is read into an existing user database, it will only overwrite identical User ID's in the current database. Otherwise, it will append additional users to the existing database.

where: *filename* is the name of the file from which or to which the user database is being written.

**Customization:** dbwrw is an application, the file itself cannot be customized.

**Example:** A customer is changing their SmartGate Server from a Microsoft Windows NT to a Sun Solaris operating system. They want to transfer their existing SmartGate user database of 1000 users to the new UNIX-based server.

1. After backing up their existing user database the administrator changes to the SmartGate Server's root directory and types:

**dbwrw -w newusrdb**

A formatted text file called newusrdb is created.

2. The administrator then transfers the text file to the new Solaris computer where SmartGate has already been installed, changes directory to the /SmartGate Server's root directory/bin, and types:

**dbwrw -r newusrdb**

## Configuration Files

The following section contains descriptions of SmartGate Server configuration files. These files contain options that you can modify to customize your SmartGate configuration.

### adm-gw.acl

Administrative Access Control List

**Used By:** SmartAdmin, sgccsrv

**Purpose:** Specifies the users who have administrative privileges to access the SmartGate Server and perform remote administration. It also specifies what level of administrative rights they have and which groups they may administer.

**NOTE:** If you are transferring the file using FTP, it must be in ASCII format (type **asc**).

**NOTE:** The adm-gw.acl file maximum line length is 256 characters.

**Location:** Installed in the SmartGate Server's root directory/etc on a UNIX-based server and SmartGate Server's root directory\data on a Microsoft Windows NT.

**Structure:** This file is a text file that contains a list of all the SmartGate administrator's User IDs. Each line contains a User ID, an access level, and the groups whom that administrator can access.

**Customization:** This is a text file which you can edit using SmartAdmin, either remotely or locally (on a Windows NT SmartGate Server). You can also modify it locally using a text editor from the server console or by running `./setup` from the `/bin` directory on a UNIX SmartGate Server. In order to perform remote administration, you must first add your SmartGate User ID to this file and assign yourself superuser privileges for all users.

There are four levels of SmartGate administrative privileges. The administrative level controls an administrator's privileges regarding users and groups.

- **Minimal** Administrators at this level can only enable/disable users and edit a user's name in the event of a name change or a typographical error. Administrators at this level may also be restricted to administering only certain groups.
- **Restricted** In addition to those rights provided at the minimal level, administrators can change authentication keys and add/edit/delete end users. Access at this level may be limited to certain groups.
- **Standard** In addition to those rights provided at the restricted level, administrators can add/edit/delete access permissions. Access at this level may be limited to certain groups.
- **Superuser** Administrators at this level have access to all settings. In addition to the privileges of the standard administrator, superusers can assign administrators, change SmartGate configuration settings, and configure the OLR, IPsec Channels, and Single Port Proxy Map files. Superusers have full access to all groups.

The `adm-gw.acl` file contains all users permitted to administer the SmartGate Server remotely, through SmartAdmin, Telnet, or a Web browser. Once you have been added to the SmartGate user database, you must add your User ID to `adm-gw.acl`.

To add yourself as an administrator to the `adm-gw.acl` file using SmartAdmin:

1. Open SmartAdmin and click the **Admin Rights** tab.
2. Click the **Add** button at the bottom of the window.
3. Enter the following information for yourself:
  - Your User ID
  - Superuser
  - Select the **Admin any group** check box

**Example:** The following is an example of the `adm-gw.acl` file:

```
Shar15838 superuser any
Patr05904 superuser any
John24685 restricted admin, finance, sales
Russ34496 minimal any
Joex39204 standard sales
Jsmith
```

## aliases.

Mail Aliases File

**Used By:** The mail program on a UNIX system.

**Purpose:** Host Configuration. Sets up e-mail addresses for any e-mail sent from SmartGate. By default, Daily and Weekly reports are sent via e-mail.

**Location:** `/etc/` directory of every UNIX-based SmartGate Server.

**Structure:** This is a one-line text file that contains the following line by default:

```
smartgate-admin: root
```

**Customization:** By default, the installation script will insert a line into `aliases.` that sends all e-mail to `root`. To send Daily and Weekly reports to an e-mail address other than `root`, you should edit this file.

**Example:**

```
smartgate-admin:<TAB>root.admin@yours.com
```

**NOTE:** Unless all three fields for `adm-gw.acl` are completed, SmartAdmin will not be available to the administrator. However, the administrator will have administrative privileges via the console.

**NOTE:** There should be at least one space or a TAB between the colon and "root."

**NOTE:** IPSec is only available with SmartGate 4.0 and later running on a Microsoft Windows NT Server.

## chantype.ini

### SmartGate Server IPSec Configuration File

**Used By:** This file is used by the `ipsec.ac1` file for obtaining configuration values specific to IPSec authentication.

**Purpose:** Contains tunnel (channel) definitions. Channel types are used to describe a certain set of security parameters to be used for a particular access permission(s). The `chantype.ini` file contains by default the definition of a channel called “encrypted.” This definition must always be in `chantype.ini`.

**Location:** Installed in the SmartGate Server’s data directory on a Microsoft Windows NT Server.

**Structure:** The `chantype.ini` file contains a “section” for each channel type. The sections will be started with “[<name>]” where <name> is the name by which the channel type will be referenced in all other data files.

Channel types are used to describe a certain set of security parameters to be used for a particular access permission. For example, one channel type might be “StrongEncryption”, which would use 3DES encryption and SHA authentication, while another might be AuthOnly, having no encryption and MD5 authentication. Each access permission definition (in the `ipsec.ac1` file) will either have a ‘channel=xxx’ clause to indicate which channel to use, or it will use the default channel (either the one with Name=encrypted, or the first channel in `chantype.ini`).

Each section will have the following *key=value* pairs.

#### Description:

Type:	ASCII String
Meaning:	A binary code for text
Default:	“”

#### SupportedEspCryptTransforms

Type:	Comma-delimited list of ASCII-decimal numbers
Meaning:	A list of all the values of <code>esp.crypt_transform</code> allowed for this channel. If no encryption is an allowed option, this list should also contain “0”

Default: "" - use default encrypt method—usually 3DES

Valid values and their meanings:

0	-	None
2	-	DES
3	-	3DES

#### **SupportedEspAuthAttributes**

Type: Comma-delimited list of ASCII-decimal numbers

Meaning: A list of all the values of `esp.auth_attribute` allowed for this channel. If no `esp-auth` is an allowed option, this list should also contain "0"

Default: "" - use default esp-auth method—usually MD5

Valid values and their meanings:

0	-	None
1	-	MD5
2	-	SHA

#### **SupportedAhAuthTransforms**

Type: Comma-delimited list of ASCII-decimal numbers

Meaning: A list of all the values of `ah.auth_transform` allowed for this channel. If no `ah-auth_transform` is an allowed option, this list should also contain "0".

Default: "" - use default ah-auth\_transform—usually MD5

Valid values and their meanings:

0	-	None
2	-	MD5
3	-	SHA

#### **SupportedAhAuthAttributes**

Type: Comma-delimited list of ASCII-decimal numbers

Meaning: A list of all the values of `ah.auth_attribute` allowed for this channel. If no `esp-auth_attribute` is an allowed option, this list should also contain "0".

**NOTE:** While it may seem that this key and the previous key are redundant, that is not 100% true. IKE defines two separate attributes, so we do the same to make sure we are able to configure any new RFC-approved AH transform that comes along.

Default:     “” - use default ah-auth\_attribute—usually MD5

Valid values and their meanings:

0	-	None
1	-	MD5
2	-	SHA

**SupportedIpCompTypes=<numeric list>**

Type:       Comma-delimited list of ASCII-decimal numbers

Meaning:    A list of all the values of IP payload compression allowed for this channel. If no `ipcomp` is an allowed option, this list should also contain “0”.

Default:     “” - use default esp auth—usually DEFLATE

Valid values and their meanings:

0	-	None
2	-	Deflate

### Customization:

This is a text file which you should edit in order to customize your system. You can edit `chantype.ini` using SmartAdmin from the IPsec channels tab, either remotely or locally. You can also modify it locally using a text editor from the server console. In order to administer remotely, you must be listed as a superuser in `adm-gw.acl`.

### Example:

```
[encrypted]
Description=Default Channel
SupportedEspCryptTransforms=3,2
SupportedEspAuthAttributes=2,1
SupportedAhAuthTransforms=0
SupportedAhAuthAttributes=0
SupportedIpCompTypes=2,0
[FastLocal]
Description=weak encryption, no auth, no compression
SupportedEspCryptTransforms=2
SupportedEspAuthAttributes=1
SupportedAhAuthTransforms=0
SupportedAhAuthAttributes=0
SupportedIpCompTypes=0
```

```
[StrongFastModem]
Description=strong encryption + ah auth + deflate
compression
SupportedEspCryptTransforms=3,2
SupportedEspAuthAttributes=0
SupportedAhAuthTransforms=2
SupportedAhAuthAttributes=3
SupportedIpCompTypes=2,0
```

## sgconf.ini

### SmartGate Server Configuration File

**Used By:** This file is used by most of the SmartGate Server components (e.g., sweb, sgate, and sgftp) for obtaining configuration values. Changes to sgconf.ini values become effective the next time SmartGate is invoked. It is not necessary to reboot the system.

**Purpose:** Dynamically controls the SmartGate Server's behavior. The sgconf.ini file provides information to SmartGate about your specific environment (e.g., where the Authentication Server resides and whether you are using Web servers that require access tickets.)

**Location:** Installed in the SmartGate Server's root directory/etc on a UNIX-based server and the SmartGate Server's root directory\data directory on a Microsoft Windows NT.

**Structure:** This file consists of comment lines and control lines. The length of each line is limited to 255 characters including spaces. Comments must be preceded by a semicolon (;). Each control line is a *name=value* pair. Name is not case sensitive.

**Customization:** This is a text file which you should edit in order to customize your system. You can edit sgconf.ini using SmartAdmin, either remotely or locally (on a Windows NT SmartGate Server). You can also modify it locally using a text editor from the server console or by running ./setup from the /bin directory on a UNIX SmartGate Server. In order to administer remotely, you must be listed as a superuser in adm-gw.acl.

For explanations of each of the configuration options available for the sgconf.ini file, see "[SmartGate Server File Option Descriptions](#)" later in this appendix.



The following is an example of an `sgconf.ini` file with only mandatory options displayed. Your file may appear different depending on which options are assigned.

### Example:

```
[sgconf]
; *****
;
; SMARTGATE SERVER CONFIGURATION FILE
;
; (sgconf.ini)
;
; The following settings are described here (in order) :
;
; authenticator (REQUIRED--see below)
; domainname (REQUIRED--see below)
; InsideIP (REQUIRED--see below)
; *****
authenticator=sample
;
; This setting is the authenticator used by clients to associate secure
; paths with this SmartGate server. This setting can be any string of
; up to 14 alphanumeric characters, but it is recommended that it be based
; on or be a derivative of your SmartGate server's host name. This
; setting was called "KeyName" and was located in the template.cat file
; in versions of SmartGate server before 2.4.
;
; THIS SETTING HAS NO DEFAULT AND IS REQUIRED FOR THE OPERATION OF SMARTGATE SERVER.
;
; Possible values: 1. the authenticator name (up to 14 alphanumeric characters)
;
; Default: there is no default--this setting is required
;
; *****
domainname=your.server.hostname.here
;
; This setting specifies the host name or IP address of your SmartGate
; server as seen from outside your firewall. It is the host name or
; IP address to which SmartPass clients will attempt to connect to when
; making secure connections to your trusted network. This setting was
; located in the template.cat file in versions of SmartGate server before 2.4.
;
; THIS SETTING HAS NO DEFAULT AND IS REQUIRED FOR THE OPERATION OF SMARTGATE SERVER.
;
; Possible values: 1. outside host name or IP address of SmartGate server
;
; Default: there is no default--this setting is required
;
; *****
InsideIP=0.0.0.0
;
; This setting specifies the IP address of your SmartGate server as seen
; from inside your firewall. It is the address from which the server will
; make proxy connections to destination servers within your trusted
; network. This setting was located in the template.cat file in versions
; of SmartGate server before 2.4.
;
; THIS SETTING HAS NO DEFAULT AND IS REQUIRED FOR THE OPERATION OF SMARTGATE SERVER.
;
; Possible values: 1. inside IP address of SmartGate server
;
; Default: there is no default--this setting is required
;
; *****
```

**Options:** See “[SmartGate Server File Option Descriptions](#)” later in this appendix for details on the options you may specify in `sgconf.ini`.

## **reginfo.dat**

SmartGate Server On-Line Registration (OLR) Configuration File.

**Used By:** Accessed by SmartPass during OLR.

**Purpose:** This file defines the data entry fields displayed to users when they perform OLR. If multiple OLR methods are being used (e.g., VONE, Entrust, Netrust, and PKI), each method can display unique data entry fields. The OLR information is read from `reginfo.dat` into an HTML file to be opened by your user’s Web browser.

**Location:** Installed in the SmartGate Server’s root directory/etc on a UNIX-based server and SmartGate Server’s root directory\data on Windows NT.

**Structure:** The first line in the `reginfo.dat` file is the server’s encryption keyname, which is used to retrieve the SmartGate’s Server’s public key. The encryption keyname is required regardless of OLR method. All subsequent lines are at the discretion of the administrator, although at least two alphanumeric data entry fields must be defined.

The default `reginfo.dat` file is displayed below:

```
encryption keyname
First Name:20:alphanum
Last Name:20:alphanum
```

The V-ONE OLR method is the default and must be placed first, hence, does not require a section header. If multiple OLR methods are being used, the section header defining the OLR method is placed before each additional set of data entry fields. For example, if the Netrust authentication method is being used in conjunction with the User ID Server (UID Server), a separate section for the Netrust OLR method must be created. Likewise, a separate section for the Entrust OLR method is needed when using Entrust authentication. You may enter up to 10 data entry fields for each OLR method.

**NOTE:** The encryption keyname must be lowercase, it must have a minimum of five and a maximum of eight characters, the first three characters must not be “key” and the first four characters must not be “test.” Digits 0-9 may also be used.

**NOTE:** The encryption keyname will be automatically inserted by the installation script.

**NOTE:** The words “Line 1 ... Line 11” do not appear in the file. They are shown here for reference purposes only.

**NOTE:** Please refer to the *SmartGate With Netrust Authentication Guide* for detailed information on SmartGate configuration options.

**NOTE:** See “[Using Entrust for User Authentication](#)” in Chapter 6, “User Authentication” for more information on Entrust authentication.

**NOTE:** For RSA SecurID and Radius authentication, the encryption keyname is displayed after the initial logon. The certificate or keyname is written to the registry.

The following sample `reginfo.dat` file, using all possible OLR methods, illustrates the file structure. An explanation of each line follows the sample.

```
Line 1    mykey
Line 2    First Name:20:alphanum
Line 3    Last Name:20:alphanum
Line 4    Phone No.:20:phone
          •    Soc.Sec.No.:11:ssn
          •    Credit Card No.:20:ccnumber
          •    Exp. Date (mmyyyy):6:numeric
          •    Street Address:60:alphanum
          •    City/State/Zip:60:anything
          •    Group:20:grouplist:gold;silver;bronze
Line 11   E-mail Address:40:anything
Line 12   [Netrust]
          •    First Name:20:alphanum
          •    Last Name:30:alphanum
          •    Group:20:grouplist:Netrust
          •    E-mail Address:40:anything
Line 17   [Entrust]
          •    First Name:20:alphanum
          •    Last Name:30:alphanum
          •    Soc.Sec.No.:11:ssn
Line 21   E-mail Address:40:anything
Line 27   [PKI]
          •    First Name:20:alphanum
          •    Last Name:30:alphanum
          •    Group:PKI, Engineering, Sales
```

- **Line 1 (Required)** This line must be the encryption keyname that was provided by you during the SmartGate Server software installation. The program will automatically enter the keyname that is reflected in `keyname.pub` and `keyname.has`.

This keyname must be unique in the Web server and database. The encryption keyname and authenticator are displayed to the end user after a successful OLR. For example:

```
<host/IP/<userid>/authenticator/<keyname>
```

- **Lines 2 and 3 (Required)** The first two OLR data entry fields make up the user's long name in the user database. The type **alphanum** is required and no spaces or special characters are allowed.

In the standard VONE OLR method section, they are labelled, "First Name" and "Last Name" by default, but they can be changed. The field names should be easy for your end users to understand. If you want to capture the full name of your end user, retain the default values. The user's long name is the first field in `reginfo.dat` followed by the second field, with a blank in between. For example:

```
users long name=John Smith
```

- **Lines 4–11 (Additional Data Entry Fields)** You can define additional information that you wish to obtain from each end user.
- **Lines 12 and 17** These lines are examples of section headers. The name in the brackets (such as Netrust) defines the OLR Method for the section following it. The default OLR Method, VONE, does not need a section header.
- **Lines 13–14 and 18–19 (Required if that OLR Method is being used)** The first two OLR data entry fields from each OLR method section will make up the user's long name in the user database. If you want to capture the full name of your end user, label them, "First Name" and "Last Name," as in the default VONE OLR method—Line 2 and 3. The type **alphanum** is required and no spaces or special characters are allowed.
- **Lines 15–16 and 20–21 (Additional Data Entry Fields)** These lines are examples of additional data entry fields for the Netrust and Entrust OLR methods. Each OLR method can have up to ten data entry fields, total.

**NOTE:** No additional data entry fields are required by the system. However, when configured, entry is mandatory. The end user will be unable to continue with OLR if information is not entered into each information field.

**NOTE:** Before modifying `reginfo.dat`, you should make at least one backup copy of it.

**Customization:** This is a text file which you can edit using SmartAdmin, either remotely or locally (on a Windows NT SmartGate Server). You can also modify it locally using a text editor from the server console or by running `./setup` from the `/bin` directory on a UNIX SmartGate Server. In order to administer remotely, you must be listed as a superuser in `adm-gw.acl`.

This file contains the formats of the data entry fields that will be displayed to your users when they perform OLR through their Web browser. You can have up to ten data entry fields in this window. The syntax for the lines that can be included in this file is shown under the heading **Structure**, above. The format for each line is:

*field\_name*:*max\_length*:*type*:*arg-list*

where

*field\_name* defines the title of the input boxes on the OLR Web form.

For example: **Phone Number**

*max\_length* defines the maximum number of characters that can be typed in the field.

*type* defines the type of data entry (valid characters) performed on the field. Nine types are available:

Type	Valid Entries
alphanum	alphabetic, 0-9
numeric	0-9
phone	0-9, comma (,) dash/hyphen (-)
grouplist	alphabetic, 0-9
ccnumber	0-9
passnum	0-9
password	alphabetic, 0-9
ssn	0-9, dash/hyphen (-)
anything	no type check

**NOTE:** *arg-list* is only used when the *type* is "grouplist."

*arg-list* currently applies only when type is **grouplist**. Each group name is delimited by semicolons.  
For example: `group:20:grouplist:gold:silver;bronze`

## netaccess.cf

Configuration file for vplug.

**Used By:** The vplug Proxy.

**Purpose:** SmartGate Configuration. The configuration file from which vplug reads its rules.

**Location:** Installed in the SmartGate Server's root directory/etc on a UNIX-based server and SmartGate Server's root directory\data on Windows NT.

**Customization:** Administrator created.

**Example:** An example of a netaccess.cf file is shown below:

```
vplug: -timeout 600
vplug: -virtdst 206.205.74.245 -virtport 389 -realdst
20.0.0.41 -realport 389
```

## net\_list

Host/Network Patterns file for vplug.

**Used By:** The vplug Proxy.

**Purpose:** SmartGate Configuration. Contains the host or network patterns from which vplug connections are allowed.

**Location:** vplug will try to open the filename as is for full paths and, if that fails, will try to open the filename relative to the SmartGate Server's root directory.

**Structure:** A *net\_list* file contains one or more host or network patterns. A host/network pattern may be an IP address or hostname. It may also contain an \* wildcard (i.e., \*.tcp.net) or use an *IP\_address:netmask* to match a single host, a subnet, or a network (i.e., 10.0.0.0:255:0.0.0).

**NOTE:** For detailed information on the netaccess.cf file, see [Chapter 10, "Using the vplug Proxy."](#)

**NOTE:** See “UID Server for On-Line Registration” in Chapter 7, “On-Line Registration Services” for a complete discussion on the UID Server.

An entry in either `netaccess.cf` or a nested `net_list` file beginning with an `@` sign will be taken as a file name from which additional host patterns will be read so if `@additional_hosts` is in `trusted_nets`, the file `additional_hosts` will be opened and any line not starting with a `#` will be taken in as one or more host or network patterns.

**Customization:** Administrator created.

**Example:** An example of a `netaccess.cf` file with a `net_list` reference in bold is shown below:

```
vplug; -virtdst 206.205.74.247 -virtport 389 -  
realdst 20.0.0.41 -realport 389 -srcaddr@jeffgood
```

**Example:** A `net_list` file called `jeffgood`:

```
206.205.74.249
```

## Optional Server Files for Enhanced Features

The following section contains descriptions of the files you may create or customize if you are implementing any of SmartGate’s enhanced features.

### Rules File

Rules File for the User ID (UID) Server

**Used By:** The UID Server

**Purpose:** Contains User IDs that you configure. During OLR, the OLR Server will assign a user the appropriate User ID from this file—instead of the random User IDs it would normally generate—based on whether the match part associated with the User ID in the Rules File matches input provided by the user.

**Location:** Specified by the `UidFile` option in the `sgconf.ini` file.

**Structure:** The Rules File is a text file consisting of a set of records, one per line. Each record has a match part and a User ID part; use a semicolon to delimit them. The match part consists of one or more *name=value* pairs; use an ampersand (&) to connect the pairs. The Rules File records do not need to be in any special order. The maximum number of records you can require of your users is limited to the total number of characters that may be on one line (511). The number of *name=value* pairs should only be as many as are necessary to guarantee a correct match.

- The *name* part and the *value* part may contain embedded spaces.
- The *name* part and the *value* part may not contain leading or trailing spaces.
- You may use leading spaces before the User ID (after the semicolon) to improve readability but the User ID itself may not contain embedded spaces.

**Customization:** You are responsible for creating the Rules File and specifying its name and location (if you are not using the default). For details on specifying the location of the Rules File, see the previous heading **Location**.

**Examples of Rules File Records:** A typical record in the Rules File might look like this:

```
First Name=Joe&Last Name=Smith&phone=555-1212
&SSN=111223333;j.smith
```

**Guidelines For Matching:** The UID Server for OLR will read the records in the Rules File sequentially from the beginning until a match is found between the *name=value* pairs in the file record and those received from your user via the OLR Server.

- All matching is case blind.
- All of the *name=value* pairs in the Rules File record must be matched by *name=value* pairs entered by your user. However, not all of the *name=value* pairs entered by your user must be matched by corresponding records in the Rules File.

**NOTE:** For group lists, the *name=value* pair should be *group=value*.



For example, you may decide that the matching will be done solely on Social Security Number. In this case, a record in the Rules File might contain:

```
SSN=111223333; user.id
```

The data received from the user might consist of other information, such as:

```
First Name=joe&Last Name=smith&phone=555-1212  
&SSN=111223333
```

The unused *name=value* pairs—in this case, *first name*, *last name*, and *phone*—are ignored.

# SmartGate Server File Option Descriptions

This section contains descriptions of the configurable options in the SmartGate Server configuration files.

## sgconf.ini Options

The following subsection contains descriptions of the options that may be specified in `sgconf.ini`. Table A-1 compares the options and their categories as described in the remote administration interface, SmartAdmin, with the actual `sgconf.ini` option names. See “Setting Configuration Options” in Chapter 5, “Using SmartAdmin,” for more information.

*Table A-1  
Comparison of Configuration Settings*

SmartAdmin Category	SmartAdmin Setting	sgconf.ini Option
Access Control	Use TCP access control	sgateacl
	Use Web access control	swebacl
	Server proxy timeout	max_quiet_time
	Web Denial Server host & port	denial_server
On-Line Registration	OLR methods	OLRMethod
	User ID Servers and ports	uid_server
	New OLR users enabled	online_reg_enable
	Netrust anonymous registration	anon_reg_allowed
	OLR data destination host and port	online_reg_service
	User ID Rules File	UidFile
	Encryption keyname	keyname (in reginfo.dat)
Logging	Reverse DNS lookups	dns_reverse
	Accounting service host & port	accounting_service
	Usage service host & port	stat_server
	Event log service host & port	event_log
	Debug reporting	debug
System Definition	SmartGate Server name	domainname
	Port list	PortList
	UDP port list	UDPPortList
	Authenticator name	authenticator
	Inside IP address	InsideIP
	SG encrypt methods	SGEncryptMethod
	Proxy encrypt methods	ProxyEncryptMethod
Authentication	Authentication methods	AuthMethod
	Remote Authentication Server & Authentication Server host	sgasrv
	Authentication client hosts	sgasrv_clients
	Auth encrypt methods	AuthEncryptMethod
	Backup server host & port	backup_userdb
	Access failure retry delay	RETRY_DELAY
	Trust CA list	TrustedCAList

SmartAdmin Category	SmartAdmin Setting	sgconf.ini Option (Contin.)
<b>Dynamic Configuration</b>	Configuration Server host & port Configuration client hosts	sgccsrv sgccsrv_clients
<b>Destination Configuration</b>	SmartGate aware services User info to Web servers Web servers requiring encrypted tickets	SmartGate_aware UserInfoToWebServer ticket_to_web_server
<b>RADIUS</b>	Backend Servers: Host Backend Servers: Secret Backend Servers: Use CHAP Backend Servers: Wait Time to live Challenge timeout	radius_authsrvn radius_authsrvn_secret radius_authsrvn_usechap radius_authsrvn_waitfor radius_ttl radius_challenge_timeout
<b>IPSEC</b>	IPSEC Server External Interface NAT Enabled NAT Network Adapter base policies	ipsec_server_extrn NAT NATNet Not in sgconf.ini— changes registry settings directly
<b>Other</b>	accesscodedaysvalid krakit_delta_days sdi_timeout sdi_ttl sgftp_port_max sgftp_port_min shim_permitexe smartwebport	AccessCodeDaysValid KraKit_Delta_Days* SDI_TIMEOUT SDI_TTL sgftp_port_max sgftp_port_min shim_permitexe SmartWebPort
<b>OLR Setup</b> tab <b>OLR Branding</b> button	Startup description Startup argument Company name Web page Street address City State Zip code Country Phone number E-mail All outside firewall	OLRStartDesc OLRStartArgs OLRCompanyName OLRWebPage OLRStreetAddress OLRCity OLRState OLRZipCode OLRCountry OLRPhoneNumber OLREmail OLRAllOutsideFirewall

\* Refer to the standalone *KRAKit Guide* for information regarding the KRAKit software.

## AccessCodeDaysValid

**Purpose:** Specifies the maximum number of days that an Access Code can age before this server will require the user to change it. If this value is zero, there is no maximum. This value is downloaded to SmartPass during Dynamic Configuration and enforced by SmartPass.

**Format:** `AccessCodeDaysValid=#days`

**Value(s):** *#days* is the maximum number of days that an Access Code will be active before the user must change it. The value will be either:

“1” to “999”  
or  
“0” —no maximum

**Default:** 0

## accounting\_service

**Purpose:** Specifies either the hostname or IP address and the port number of the accounting service that will be sent SmartGate session accounting information at the end of each SmartGate session.

The following information is sent:

- IP address of SmartPass
- User ID of the user originating the session
- Session start time (seconds since epoch)
- Session end time (seconds since epoch)
- Destination host
- Destination port
- Server to client number of bytes sent
- Client to server number of bytes sent
- The word “misc”

**Format:** `accounting_service=host:port`

**Value(s):** *host* is either the hostname or IP address of the computer which will serve as the accounting service host.

*port* is the port number of the accounting service host.

*port* defaults to 4839; this is recommended, but you may modify it if needed.

**NOTE:** The default heading indicates the assigned default values as specified in `sgconf.ini`. You should ensure that these option names and their default values remain present in the file or that you change the default values for your specific implementation.

**NOTE:** For more information regarding the Accounting Service, see “[Accounting Service](#)” in Appendix B, “Services.”

**NOTE:** Please refer to the *SmartGate With Netrust Authentication Guide* for detailed information on SmartGate configuration options.

**WARNING!** This option is required. You must ensure that the appropriate option value is present in the `sgconf.ini` file.

**Default:** None. The SmartGate Server will not send accounting information if `accounting_service` is not specified.

**Example:** `accounting_service=123.123.123.123:4839`

### **anon\_reg\_allowed**

**Purpose:** Specifies if SmartGate allows for anonymous registration when using the Netrust authentication method.

When using Netrust authentication, SmartGate administrator's have the option of having their end users register anonymously or having them perform OLR, in which case they must disallow anonymous registration and set up a UID Server. The `anon_reg_allowed` option is used in conjunction with the `OLRMethod` option in `sgconf.ini`, and in addition to the `reginfo.dat` file, to specify Netrust authentication using a UID Server.

**Format:** `anon_reg_allowed=yes|no`

**Value(s):** `yes` allows for anonymous registration.

`no` disallows anonymous registration.

**Default:** Yes; anonymous registration is allowed for Netrust authentication.

#### **Examples:**

`anon_reg_allowed=yes`

### **authenticator**

**Purpose:** Identifies the smart card **Authenticator name**, which is stored on the user's smart card. It can be up to 14 alphanumeric characters in length.

**Format:** `authenticator=name`

**Value(s):** `name` is the smart card Authenticator name to be stored on the user's physical or virtual smart card with the user's authentication key.

**Default:** None

**Example:** `authenticator=yourkey`

## AuthEncryptMethod

**Purpose:** Specifies the method in which SmartGate Authentication traffic will be encrypted. The authentication traffic includes all traffic between the Authentication Server (*sgasrv*) and its clients (*sgasrv\_client*) and the Configuration Server (*sgccsrv*) and its clients (*sgccsrv\_client*). Available methods include plain (no encryption), DES, and 3DES. DES is the default method.

**Format:** `AuthEncryptMethod=plain|DES|3DES`

**Value(s):** *plain* no encryption is used.

*DES* is the standard 56-bit [DES encryption](#) method.

*3DES* is the 3-key [Triple DES encryption](#).

**Default:** DES

**Example:** `AuthEncryptMethod=DES`

## AuthMethod

**Purpose:** Specifies the method used to authenticate users. VONE is the standard default method and should not be removed. The value string is displayed as a comma-delineated list.

**Value(s):** *VONE* is the standard authentication method used by the SmartGate software.

**Default:** VONE

## backup\_userdb

**Purpose:** Specifies either the hostname or IP address and the port number of the [backup host](#) on which the redundant user database (*sgrdb*) will reside. All changes to the user database on the Authentication Server will be mirrored on the backup host. This feature is optional. Unless a backup host has been set up and this option is specified, no backup will be created.

**Format:** `backup_userdb=host:port`

**Value(s):** *host* is either the hostname or IP address of the backup host. It should point to a remote host; not localhost.

*port* is the port number of the backup host. While the port number is configurable, the default of 3901 is recommended.

**NOTE:** Triple DES encryption (3DES) is only available with SmartGate 2.6 and SmartPass 3.3 and later versions. If previous versions are used for either server or client, this setting will be ignored.

**NOTE:** For more information see “Switching Between the Production and Debug Logs” in Appendix B, “Services.”

**NOTE:** See “Denial Server” in Appendix B, “Services,” for more information.

**Default:** None. If this option is omitted from `sgconf.ini`, this feature is not used.

**Example:** `backup_userdb=222.111.111.111:3901`

## debug

**Purpose:** Switches between production and debug logs by specifying what level, if any, of debug information will be written, in addition to the default standard logging information. The production and debug logging information is written to the Microsoft Windows NT Event Viewer or to `/var/log/smartgate` on a UNIX-based SmartGate Server. The primary purpose of this setting is to obtain additional log messages about the operations of the SmartGate Server.

**Format:** `debug=level`

**Value(s):** *level* is either:

- 1 only error messages will be written to the log. Do not use this setting unless directed by a qualified technical support person as it increases performance at the expense of security.
- 0 error messages and user connection information is written to the log. This is the same as standard logging.
- 1 or higher an increasing amount (equal to the level number) of debugging information will be written to the log.

**Default:** 0

**Example:** `debug=1`

## denial\_server

**Purpose:** Specifies either the hostname or IP address and the port number of the Denial Server. If set, the Denial Server is called before every Web access through the Web Proxy (`sweb`). The Denial Server receives information about each Web request and returns a “GRANT,” “DENY,” or “PASS” (refer to `sweb.acl`) response. This feature is optional.

**Format:** `denial_server= host:port`

**Value(s):** *host* is the hostname or IP address of the computer where your Denial Server will reside.

*host* will default to 127.0.0.1 if not present.

*port* is the port number where your Denial Server will reside.

**Default:** blank—there is no Denial Server.

## dns\_reverse

**Purpose:** Specifies whether the server should attempt to do a reverse [Domain Name Service \(DNS\)](#) hostname lookup on all client connections.

**Format:** `dns_reverse=yes|no`

**Value(s):** *yes* A DNS reverse lookup is attempted

*no* A DNS reverse lookup is not attempted and the client IP address is written to  
/var/log/smartgate with a hostname of  
“unknown.”

**Default:** yes

**Example:** `dns_reverse=yes`

## domainname

**Purpose:** Specifies the [Fully Qualified Domain Name \(hostname\)](#) or [IP address of your SmartGate Server](#) (specifically, where your OLR Server resides). This option will be configured during installation of the SmartGate Server software. For licensing reasons, please make certain that your domainname is unique. Do not use a generic name like “SGServer.”

**Format:** `domainname=host`

**Value(s):** *host* is the hostname or IP address of the computer which will serve as the OLR Server.

**Default:** None. However, the installation script will attempt to obtain the appropriate information and insert it as a default value.

**Programmatic:** None

**Example:** `domainname=smartgate.your.com`

**NOTE:** If `dns_reverse` is set to **no** only IP addresses (no hostnames) can be used in the `sgasrv`, `sgasrv_clients`, `sgccsrv`, and `sgccsrv_clients` settings.

**WARNING!** This option is required. You must ensure that the appropriate option value is present in the `sgconf.ini` file.

**WARNING!** To use IPsec features or UDP broadcasting (`UDPPortList`), the IP address **MUST** be used **NOT** a hostname for the domainname.



**WARNING!** This option is required. You must ensure that the appropriate option value is present in the `sgconf.ini` file.

**NOTE:** `InsideIP` accepts a comma separated list. The server will broadcast with each of the IP addresses (together with the `UDPPort`) to all interfaces and will then wait for the return packet.

## **event\_log**

**Purpose:** Specifies either the hostname or IP address and the port number of an optional UDP Server which will be sent a copy of the same log entries as the event log flat file. The event log file, `sgevent.log`, is located in `/SmartGate Server root directory/etc` on a UNIX-based server or to `SmartGate Server root directory\data` on a Microsoft Windows NT server.

**Format:** `event_log=host:port`

**Value(s):** *host* is either the hostname or IP address of the computer which will serve as the host of the additional copy of the event log.

*port* is the port number of the event log host.

**Default:** None. The SmartGate Server will not send event log information if `event_log` is not specified.

**Example:** `event_log=222.111.111.111:2080`

## **InsideIP**

**Purpose:** Specifies the IP address assigned to the inside **network adaptor card** on the SmartGate Server. This address is used to configure the Remote Administrator.

**Format:** `InsideIP=host`

**Value(s):** *host* is the IP address assigned to the inside network adaptor card on the SmartGate Server.

**Default:** The installation script will attempt to obtain the appropriate information and insert it as a default value.

**Programmatic:** None

**Example:** `InsideIP=222.111.111.111`

## **ipsec\_server\_extrn**

**Purpose:** Specifies the external interface (hostname or IP address) of the machine where you want your IPsec request to be sent. Normally this would be the SmartGate Server. However, if you are using a firewall with serial topology (i.e., SmartGate with two interfaces placed directly behind a firewall in serial—thus all traffic passes through both firewall and SmartGate), it should be set to the external interface of the firewall.

**Format:** `ipsec_server_extrn=host`

**Value(s):** *host* is the hostname or IP address assigned to the external interface on the SmartGate Server or the firewall, depending on your network configuration.

**Default:** The installation script will attempt to obtain the SmartGate Server's external interface and insert it as a default value.

**Programmatic:** None

**Example:** `ipsec_server_extrn=222.111.111.111`

## Krakit\_Delta\_Days

**Purpose:** Specifies the number of days that the KRAKit Server will be allowed to search for deleted user keys. The valid range is 1 to 1000 and the default is 15 days.

**Format:** `Krakit_Delta_Days=days`

**Value(s):** *days* is the number of days the KRAKit Server can search for deleted keys.

**Default:** 15

**Programmatic:** None

**Example:** `Krakit_Delta_Days=30`

## max\_quiet\_time

**Purpose:** Specifies the number of seconds of idle time before a SmartGate Server (proxy) will timeout and close its connection to SmartPass.

**Format:** `max_quiet_time=seconds`

**Value(s):** *seconds* is the number of seconds to maintain the connection. The number may be up to 8 digits.  
where "0" (zero) is equal to no timeout.

**Default:** 300 seconds (5 minutes)

**Example:** `max_quiet_time=600`

## NAT

**Purpose:** Specifies if the IPsec Server needs to do NAT or not. NAT is disabled by default. If NAT is enabled, you must specify a range of addresses to be used for address translation in the setting NATNet.

**NOTE:** For more information on KRAKit, see the standalone *KRAKit Guide*.

**Format:** NAT=*yes|no*

**Value(s):** *yes* The IPsec Server will be doing NAT.

*no* The IPsec Server will not be doing NAT.

**Default:** no

### NATNet

**Purpose:** If the NAT Enabled checkbox is selected, this setting specifies the address range that the IPsec Server can use for address translation.

**Format:** NATNet=*networks*

**Value(s):** *networks* A list of networks—a range of addresses that the IPsec Server can use for address translation.

**Default:** None

**Example:** NATNet=10.0.0.10\24,10.0.0.10\9

### OLRAIIOutsideFirewall

**Purpose:** Specifies if all users are outside the [firewall](#).

**Format:** OLRAIIOutsideFirewall=*yes|no*

**Value(s):** *yes* None of your users have SmartPass behind a firewall. The proxy configuration box on the OLR Web page will not be displayed.

*no* At least some of your users may have SmartPass behind a firewall. The proxy configuration box on the OLR Web page will be displayed.

**Default:** no

### OLRCity

**Purpose:** Specifies company address information displayed on the OLR Web page.

**Format:** OLRCity=*city*

**Value(s):** *city* is the city where your company resides.

**Default:** None

## OLRCompanyName

**Purpose:** Specifies company information displayed on the OLR Web page.

**Format:** OLRCompanyName=*name*

**Value(s):** *name* is your company name.

**Default:** None

## OLRCountry

**Purpose:** Specifies company address information displayed on the OLR Web page.

**Format:** OLRCountry=*country*

**Value(s):** *country* is the country where your company resides.

**Default:** None

## OLREmail

**Purpose:** Specifies company address information displayed on the OLR Web page.

**Format:** OLREmail=*email*

**Value(s):** *email* is your company's e-mail address.

**Default:** None

## OLRPhoneNumber

**Purpose:** Specifies company address information displayed on the OLR Web page.

**Format:** OLRPhoneNumber=*phone*

**Value(s):** *phone* is your company's phone number.

**Default:** None

## OLRStartArgs

**Purpose:** Specifies the location of the Web page you want the end user to see when they start SmartPass using their desktop icon.

**Format:** OLRStartArgs=*location*

**Value(s):** *location* is the command to launch their Web browser and the URL to which the desktop SmartPass icon is connected.

**Default:** None

**Example:** OLRStartArgs=-h http://www.vone.com/

### OLRStartDesc

**Purpose:** Specifies the title you want to appear under the SmartPass icon that will be placed on your end user's desktop after they register.

**Format:** OLRStartDesc=*title*

**Value(s):** *title* is the title on your desktop SmartPass icon.

**Default:** None

### OLRState

**Purpose:** Specifies the U.S. state displayed on the company's OLR Web page.

**Format:** OLRState=*state*

**Value(s):** *state* is the state where your company resides.

**Default:** None

### OLRStreetAddress

**Purpose:** Specifies company address information displayed on the OLR Web page.

**Format:** OLRStreetAddress=*street*

**Value(s):** *street* is your company's street address.

**Default:** None

### OLRWebPage

**Purpose:** Specifies company Web address or URL information displayed on the OLR Web page.

**Format:** OLRWebPage=*URL*

**Value(s):** *URL* is your company's Web home page.

**Default:** None

### OLRZipCode

**Purpose:** Specifies company zip code information displayed on the OLR Web page.

**Format:** OLRZipCode=*zip*

**Value(s):** *zip* is your company's zip code.

**Default:** None

## OLRMethod

**Purpose:** Specifies the possible methods—V-ONE, NETRUST, or ENTRUST—used to add users during OLR. V-ONE is the standard default method and should not be removed. The value string is displayed as a comma-delineated list.

**Format:** `OLRMethod=V-ONE,NETRUST,ENTRUST,PKI`

**Value(s):** *V-ONE* is the standard method used by the SmartGate software to add users during OLR.

*NETRUST* is the method used to add users during OLR if the Netrust authentication method is being used.

*ENTRUST* is the method used to add users during OLR if the Entrust authentication method is being used.

*PKI* is the method used to add users during OLR if the PKI authentication method is being used.

**Default:** VONE

## online\_reg\_enable

**Purpose:** Specifies that users will be enabled immediately after registering via OLR.

**Format:** `online_reg_enable=yes|no`

**Value:** *yes* The user is enabled immediately after registering via OLR.

*no* The user is not enabled immediately after registering via OLR. You must enable the user account using the command line `sgadm` or remote administration.

**Default:** No; the user account must be enabled by the SmartGate administrator.

**Example:** `online_reg_enable=yes`

**NOTE:** Please refer to the *SmartGate With Netrust Authentication Guide* for detailed information on SmartGate configuration options.

**NOTE:** See “[Using Entrust for User Authentication](#)” in Chapter 6, “User Authentication,” for detailed information on Entrust authentication.

**NOTE:** The `online_reg_enable` option does not apply to RADIUS, SecurID, and Netrust (when `anon_reg_allowed` is set to **yes**) authentication, because the end user does not perform OLR.

**NOTE:** You may specify any IP address and port number, although port number 3839 is recommended.

## online\_reg\_service

**Purpose:** Specifies either the hostname or IP address and the port number of the OLR Activity Recording Service (ARS) that will receive OLR activity information at the end of each OLR session.

The information that the user enters on the screen during OLR is stored locally in the `sgreg usr` file. At the end of each OLR session, the OLR Server generates a record for the user. It writes the record to `sgreg usr` and also sends the following data to your OLR Activity Recording Service if this option is set:

- The date and time the user was registered (local to the OLR Server).
- The User ID and OLR fields of the registered user.

**Format:** `online_reg_service=hostport`

**Value(s):** *host* is either the hostname or IP address of the computer which will serve as the ARS host.  
*port* is the port number of the ARS host.

**Default:** No default. The OLR Server will not send this information if `online_reg_service` is not specified.

**Example:** The following extract from the `sgreg usr` file shows an example of the format in which the data is sent to the specified ARS. In all cases, the data and time that the user was registered are included in the data stream. The remaining fields and field labels will vary depending on what information you have requested from your end users in the `reginfo.dat` file. Fields are delimited by ampersands (&). (You must ensure that your ARS expects the data to arrive in this format.) Line numbers in the file are not sent.

```
1: DateTime=Tue May 27 15:02:08 1997&userId=
johns1234ab&First_Name=John&Lastname=Smith&
Email=john@xyz.com&
```

## Port List

This setting allows the SmartGate Server administrator to set up the server to listen on multiple ports. This allows end users to try to connect using ports other than the standard single port proxy port 3845 to navigate through firewalls.

**Format:** `PortList=port,port`

**Value(s):** *port* is the port number that the server will listen.

**Default:** 3845,443,80

## ProxyEncryptMethod

**Purpose:** This setting defines the complete set of encryption methods available to be used for proxy data packets which convey end user data between client and server. This setting is communicated once at the initial connection of a SmartGate session and remembered by the client for its session duration. The value string is displayed as a comma-delineated list of available methods; the preferred method will be listed first.

**Format:** `ProxyEncryptMethod=3DES,DES,RC4`

**Value(s):** *3DES* is the 3-key Triple DES encryption. This value will allow for Triple DES encryption to be used if it is supported by both client and server.

*DES* is the standard 56-bit DES encryption method.

*RC4* is the 40-bit RC4 encryption method.

**Default:** DES

**Example:** `ProxyEncryptMethod=3DES,DES,RC4`

## radius\_authsrv[1...5]

**Purpose:** Specifies the FQDN or IP address of the RADIUS Backend Server and backup RADIUS Servers (up to 5). If [RADIUS authentication](#) is being used, at least one value must be specified as the RADIUS Backend Server. Additional backup servers are optional.

**Format:** `radius_authsrvn=host`

**NOTE:** Triple DES encryption (3DES) is only available with SmartGate 2.6 and SmartPass 3.3 and later versions. If previous versions are used for either server or client, this setting will be ignored.



**Value(s):** *n* is a number between 1 and 5, starting at 1, which specifies a RADIUS Backend Server.

*host* is either the FQDN or IP address of the computer that will serve as the RADIUS Backend Server.

**Default:** None

**Example:**

```
radius_authsrv1=10.0.0.225  
radius_authsrv2=10.0.0.226  
radius_authsrv3=10.0.0.227
```

### **radius\_authsrv[1...5]\_secret**

**Purpose:** Specifies the shared secrets for the RADIUS Backend Server and each of its backups (up to 5). If RADIUS is being used, at least one value must be specified as the RADIUS Backend Server. Additional backup servers are optional.

Each RADIUS Backend Server must be configured with its corresponding shared secret code. See your RADIUS documentation for further information.

**Format:** **radius\_authsrv*n*\_secret=code**

**Value(s):** *n* is a number between 1 and 5, starting at 1, which specifies a specific RADIUS Backend Server.

*code* is a shared secret code between the SmartGate Server running the RADIUS module and the RADIUS Backend Server.

**Default:** None

**Example:**

```
radius_authsrv1_secret=ltzieojh54343  
radius_authsrv2_secret=ol34kduf67  
radius_authsrv3_secret=dk00rk56
```

## radius\_authsrv[1...5]\_usechap

**Purpose:** Specifies whether your RADIUS Backend Server is using CHAP authentication for its users. For each of these servers, the RADIUS module on the SmartGate Server will simulate a CHAP exchange and send a CHAP-Password value instead of the normally hashed User-Password attribute. If RADIUS is being used, each of the servers are defaulted to “no.”

**Format:** radius\_authsrv*n*\_usechap=*yes|no*

**Value(s):** *n* is a number between 1 and 5, starting at 1, which specifies a specific RADIUS Backend Server.

*yes* if this RADIUS Backend Server is using CHAP authentication.

*no* if this RADIUS Backend Server is not using CHAP authentication.

**Default:** None

### Example:

```
radius_authsrv1_usechap=yes
radius_authsrv2_usechap=yes
radius_authsrv3_usechap=no
```

## radius\_authsrv[1...5]\_waitfor

**Purpose:** Specifies the wait time (in seconds) for requests made to the RADIUS Backend Server before it times out. Network factors may prevent certain servers from responding as quickly as they should. The default is 120 seconds for each server with a maximum of 32767 seconds.

**Format:** radius\_authsrv*n*\_waitfor=*seconds*

**Value(s):** *n* is a number between 1 and 5, starting at 1, which specifies a specific RADIUS Backend Server.

*seconds* is the number of seconds a request to the RADIUS Backend Server will be “waited for” before it times out.

**Default:** 120 seconds

### Example:

```
radius_authsrv1_waitfor=120
radius_authsrv2_waitfor=60
radius_authsrv3_waitfor=32767
```

**NOTE:** Do not use commas in the radius\_authsrv[1...5]\_waitfor setting.

## **radius\_challenge\_timeout**

**Purpose:** Specifies the number of minutes that a RADIUS challenge dialog box will remain on the screen before it times out. The valid range is 1 to 30 minutes and the default is 5 minutes.

**Format:** `radius_challenge_timeout=minutes`

**Value(s):** *minutes* is the number of minutes of response time.

**Default:** 5

**Example:** `radius_challenge_timeout=3`

## **radius\_ttl**

**Purpose:** Specifies the number of minutes that a RADIUS authentication will remain valid before another authentication is required. The valid range is 1 to 1440 minutes and the default is 30 minutes.

**Format:** `radius_ttl=minutes`

**Value(s):** *minutes* is the number of minutes before authentication is again required.

**Default:** 30

**Example:** `radius_ttl=40`

## **RETRY\_DELAY**

**Purpose:** Specifies the minimum number of seconds allowed between unsuccessful user login attempts. If this is set, the user will be forced to wait the specified number of seconds before trying again after an unsuccessful attempt. If this is not set or is set to zero, the user's ID will be locked after three consecutive unsuccessful attempts.

**Format:** `RETRY_DELAY=seconds`

**Value(s):** *seconds* is the minimum seconds between unsuccessful login attempts.

where "0" (zero) locks the user's ID after three consecutive unsuccessful login attempts.

**Default:** 0

**Example:** `RETRY_DELAY=30`

## SDI\_TIMEOUT

**Purpose:** Specifies the number of minutes that an end user, when using [RSA SecurID authentication](#), will be allowed before responding to a **Next Tokencode** or **New Pin Code** dialog box. The valid range is 1 to 30 minutes and the default is 3 minutes.

**Format:** `SDI_TIMEOUT=minutes`

**Value(s):** *minutes* is the number of minutes of response time.

**Default:** 3

**Example:** `SDI_TIMEOUT=60`

## SDI\_TTL

**Purpose:** Specifies the number of minutes for which a RSA SecurID authentication will remain valid before another authentication is required. The valid range is 1 to 1440 minutes and the default is 30 minutes.

**Format:** `SDI_TTL=minutes`

**Value(s):** *minutes* is the number of minutes before authentication is again required.

**Default:** 30

**Example:** `SDI_TTL=120`

## sgasrv

**Purpose:** Specifies either the hostname or IP address of your [Authentication Server](#). This setting is used by the SmartGate proxies when they are not on the same host as the Authentication Server. Change this setting only if you have installed your Authentication Server on a computer other than the SmartGate Server. In conjunction with the `sgasrv_clients` setting, this allows more than one SmartGate Server to share one Authentication Server.

If you are using the `sgasrv` setting, you must also configure the `sgconf.ini` file on the computer where the Authentication Server resides to recognize the SmartGate Server(s) connecting to it. Use the `sgasrv_clients` setting on that computer.

**NOTE:** Do not use commas in the `SDI_TTL` setting.

**NOTE:** The Authentication Server always listens to port 3838 for requests from the SmartGate Server.

**NOTE:** If `dns_reverse` is set to **no** only IP addresses (no hostnames) can be used in the `sgasrv_clients` setting.

**Format:** `sgasrv=host`

**Value(s):** *host* is the IP address of the computer where your Authentication Server is installed.

**Default:** The default of `127.0.0.1` (`localhost`) and port `3838` are fixed values in the standard `sgconf.ini` file. However, if the Authentication Server resides on a different computer, use this option to tell `sgate` and `sgftp` where the Authentication Server is located.

**Example:** `sgasrv=111.222.222.111`

### **sgasrv\_clients**

**Purpose:** Specifies a list of hostnames or IP addresses, in addition to `localhost`, to which the Authentication Server allows connection. The Authentication Server only accepts requests from the hosts specified in `sgasrv_clients` and `localhost`.

Use this option if the Authentication Server is being shared by other SmartGate Servers. The `sgasrv_clients` setting is defined on the computer where the Authentication Server resides.

**Format:** `sgasrv_clients=host1, host2,..., hostn`

**Value(s):** *host1, host2, ...* is a list of the hostnames or IP addresses (in addition to `localhost`) of the SmartGate Servers from which the Authentication Server should accept requests. The number of hosts you may specify is limited to the maximum line length (255).

**Default:** `localhost` is always assumed.

**Example:**

`sgasrv_clients=222.111.111.111, www.ooo.com`

## sgateacl

**Purpose:** Specifies whether `sgate.acl` will be used to grant or deny users' permissions to access services through the generic (`sgate`), FTP (`sgftp`), or Oracle (`sgora`) proxies. If you set this to **no**, then all requested client connections will be allowed. It is recommended that you do not change this setting.

**Format:** `sgateacl=yes|no` (lowercase)

**Value(s):**

- yes* Turns on access control. If `sgateacl=yes`, `sgate`, `sgftp`, and `sgora` need access control through the Authentication Server for any accesses after the user is authenticated.
- no* Access control is not used for your E-mail, FTP, Oracle, or Telnet Services. If `sgateacl=no`, `sgate` and `sgftp` will only authenticate users.

**Default:** `yes`

**Example:** `sgateacl=yes`

## sgccsrv

**Purpose:** Specifies either the hostname or IP address and the port number of your Dynamic Configuration Server. Change this setting only if you have installed your Dynamic Configuration Server on a computer other than the SmartGate Server.

If you are using the `sgccsrv` setting, you must also configure the `sgconf.ini` file on the computer where the Dynamic Configuration Server resides to recognize the SmartGate Server(s) connecting to it. Use the `sgccsrv_clients` setting on that computer.

**Format:** `sgccsrv=host:port`

**Value(s):**

- host* is either the hostname or IP address of the computer which will serve as your Dynamic Configuration Server.
- port* is the port number on which the Dynamic Configuration Server is listening. Port number 3843 is recommended.

**NOTE:** When `sgateacl=yes`, you must use Dynamic Configuration rather than Manual Setup for setting secure pathways.

**NOTE:** The Dynamic Configuration Server must reside on the same computer as the Authentication Server (`sgasrv`).

**NOTE:** If `dns_reverse` is set to **no** only IP addresses (no hostnames) can be used in the `sgccsrv` setting.

**NOTE:** If `dns_reverse` is set to **no** only IP addresses (no hostnames) can be used in the `sgccsrv_clients` setting.

**NOTE:** Triple DES encryption (3DES) is only available with SmartGate 2.6 and SmartPass 3.3 and later versions. If previous versions are used for either server or client, this setting will be ignored.

**Default:** The default of 127.0.0.1 (localhost) and port number 3843 are fixed values in the standard `sgconf.ini` file. Usually, you should retain these values. However, if you are distributing your SmartGate System across multiple processors, you will need to change them.

**Example:** `sgccsrv=111.0.0.1:3843`

### **sgccsrv\_clients**

**Purpose:** Specifies a list of hosts (hostnames or IP addresses and port numbers) to which the Dynamic Configuration Server allows connection. The Dynamic Configuration Server only accepts requests from localhost and the hosts specified in the `sgccsrv_clients` list.

Use this option if the Dynamic Configuration Server is being shared by other SmartGate Servers. The `sgccsrv_clients` setting is defined on the computer where the Dynamic Configuration Server resides.

**Format:** `sgccsrv_clients=host1, host2,... hostn`

**Value(s):** `host1, host2 , ...` is a list of hostnames or IP addresses (in addition to localhost) of the SmartGate Servers from which the Dynamic Configuration Server should accept requests. The number of hosts you may specify is limited to the number of characters that will fit on one line (255).

**Default:** localhost is always assumed.

**Example:** `sgccsrv_clients=222.111.111.222, host123`

### **SGEncryptMethod**

**Purpose:** This setting defines the complete set of encryption methods available to be used for OLR, Time Server, Dynamic Configuration, and Authentication. This setting is communicated once at the initial connection of a SmartGate session and remembered by the client for its session duration. The value string is displayed as a comma-delineated list of available methods; the preferred method will be listed first.

**Format:** `SGEncryptMethod=3DES,DES`

**Value(s):** *3DES* is the 3-key Triple DES encryption. This value will allow for Triple DES encryption to be used if it is supported by both client and server.

*DES* is the standard 56-bit DES encryption method.

**Default:** DES

**Example:** `SGEncryptMethod=3DES,DES`

### **sgevent\_logging**

**Purpose:** Turns off SmartGate event logging entirely. If event logging is turned off, no messages will be sent to the event log. This option is used exclusively to increase performance at the expense of security. Event logging is written to the `sgevent.log` file in `/SmartGate Server root directory/etc` on a UNIX-based server or to `SmartGate Server root directory\data` on a Microsoft Windows NT server.

**Format:** `sgevent_logging=1|0`

**Value(s):** 1 is equal to on

0 is equal to off

**Default:** 1 (event logging is on)

**Example:** `sgevent_logging=0` (event logging is turned off)

### **sgftp\_port\_max**

**Purpose:** Specifies the maximum TCP port number to be used by the FTP Proxy for data transfers. This setting is used in conjunction with `sgftp_port_min` to limit the ports used by the FTP Proxy. The minimum port setting must be lower than the maximum port setting and the difference should be at least 10 ports.

**Format:** `sgftp_port_max=number`

**Value(s):** *number* is the maximum TCP port number that the FTP Proxy can use for data transfers.

**Default:** none

**WARNING!** V-ONE does not recommend turning off event logging. This option should only be used under direct instructions from a qualified technical support person.



## sgftp\_port\_min

**Purpose:** Specifies the minimum TCP port number to be used by the FTP Proxy for data transfers. This setting is used in conjunction with `sgftp_port_max` to limit the ports used by the FTP Proxy. The minimum port setting must be lower than the maximum port setting and the difference should be at least 10 ports.

**Format:** `sgftp_port_min=number`

**Value(s):** *number* is the minimum TCP port number that the FTP Proxy can use for data transfers.

**Default:** none

## shim\_permitexe

**Purpose:** This setting is the list of client computer executables that will bypass the blocking shim when accessing the Windows socket library (`winsock` or `wsock32`). If you are not using the blocking shim, this setting is ignored.

**Format:** `shim_permitexe=executables`

**Value(s):** *executables* is a list of executables separated by commas.

**Default:** none

## SmartGate\_aware

**Purpose:** Specifies a list of services that are ready to receive a SmartGate end user's information after the SmartGate Server has authenticated the user. The SmartGate Server will send the user's information only to those services specified in the `SmartGate_aware` list.

**Format:** `SmartGate_aware=[protocol:]host:port,[protocol:]host:port,...`

**Value(s):** *protocol* is one of the following values:

**VONE:** the default protocol

**VPOP:** mail type of service

**nothing:** defaults to VONE

*host* is either the hostname or IP address of the specified SmartGate Aware Server.

*port* is the port number of the specified SmartGate Aware Server.

**Default:** None. If this option is omitted from `sgconf.ini`, this feature is not used.

**Example:**

```
SmartGate_aware=VONE:www.ooo.com,  
vpop:222.111.111.111:1234,testserv1:321
```

## SmartWebPort

**Purpose:** Specifies the port number on which SmartPass should listen for Web connections. The default is 2080 and should not be changed.

**Format:** `SmartWebPort=port`

**Value(s):** *port* is the port number SmartPass will listen on for Web connections.

**Default:** 2080

## stat\_server

**Purpose:** Specifies either the hostname or IP address and port number of the statistics service to which the SmartGate Server will send session start and end log entries. They can be used to create statistics and reports on user sessions.

**Format:** `stat_server=host:port`

**Value(s):** *host* is the hostname or IP address.  
*port* is the port number of the service.

**Default:** None. The SmartGate Server will not send statistical information if `stat_server` is not specified.

## swebacl

**Purpose:** Lets you specify whether access control is to be used for Web services. It is recommended that you do not change this setting.

**Format:** `swebacl=yes|no`

**Value(s):** *yes* Turns on access control for your Web service.  
*no* Access control is not used for your Web service.

**Default:** yes

**Example:** `swebacl=yes`

**NOTE:** When `swebacl=yes`, you must use Dynamic Configuration rather than Manual Setup for setting secure pathways.

**NOTE:** Your Web server needs a CGI routine to decrypt the ticket and send a challenge back to the SmartGate Server.

## ticket\_to\_web\_server

**Purpose:** In conjunction with the `sgkeys` file, this setting lets you specify whether the SmartGate Server is to send an encrypted ticket to any of your Web servers and the location of those Web servers.

**Format:** `ticket_to_web_server=host1,host2,...`

**Value(s):** *host* is the hostname or IP address of each Web server for which you want authentication performed using `sgkeys` as an encrypted ticket.

**Default:** None. If this option is not specified, no ticket is sent. The SmartGate Server will send a ticket only to those Web servers specified in the `ticket_to_web_server` list.

**Example:**

```
ticket_to_web_server=www.ooo.com,222.111.11.111
```

## TrustedCAList

**Purpose:** This setting specifies the level of verification when using PKI authentication. If this value is set to **yes** and you have added CA certificates to the trusted CA list using the `certmanager` program, then the SmartGate PKI authentication server will check the validity of the date of the user certificate, verify that the certificate is the same as the original OLR PKI certificate, and verify the user certificate by one of the trusted CAs in the list.

**Format:** `TrustedCAList=yes/no`

**Value(s):** *yes* the signature of the certificate, that it is the same certificate used for OLR, and the validity of the date of the certificate are checked.

*no* only the date of the user certificate is checked.

**Default:** `no`

**Example:** `TrustedCAList=yes`

**NOTE:** The validity date check is to verify the “not before date” and the “not after date.”

## UDPPortList

This setting allows the SmartGate Server administrator to set up the server to look up a UDP port list on an `sgate` service.

**Format:** `UDPPortList=port,port`

**Value(s):** *port* is the port number that the server will listen on.

**Default:** no default

## UidFile

**Purpose:** Specifies the location (full path name) of the [Rules File](#) for use with the optional User ID (UID) Server. This setting is used in conjunction with the `uid_server` setting.

**Format:** `UidFile=location`

**Value(s):** *location* name and full path name of the Rules File.

**Default:** None. Unless both `UidFile` and `uid_server` are specified, the UID Server is not used.

**Example:**

```
UidFile=/SmartGate Server's root directory/rules001
```

## uid\_server

**Purpose:** Specifies either the hostname or IP address and the port number of the optional UID Server (`sguidsrv`). Separate UID Servers can be configured for the Entrust or Netrust authentication methods. This setting is used in conjunction with the `UidFile` setting which specifies the location of the Rules File. The UID Server is sent the end user's OLR information and must return either a User ID to be assigned to the user or a reason for rejection. Either the SmartGate UID Server may be used to create the User ID or the customer may write their own.

**Format:** `uid_server=host:port`  
`uid_server[Netrust]=host:port`  
`uid_server[Entrust]=host:port`

**Value(s):** *host* is the hostname or IP address of the UID Server. If the UID Server resides on the same computer as your SmartGate Server, set *host* to `localhost`.

*port* is the port number of the UID Server. The default port number for the SmartGate UID Server is 3846.

**NOTE:** In order for the Citrix ICA Client "Auto-Locate" feature to work, you must create a TCP Access for 255.255.255.255 Client port=1604 Server port=2023 Destination port=1604.

**NOTE:** If you use the UID Server, you must use it exclusively; you cannot combine both the use of the UID Server and automatic User ID generation in OLR sessions.

**NOTE:** For more information on the UID Server, see "[UID Server for On-Line Registration](#)" in Chapter 7, "On-Line Registration Services."

**NOTE:** If you must change the default port number of the SmartGate UID Server, see "[Setting Up Your SmartGate UID Server](#)" in Chapter 7 "On-Line Registration Services" for detailed instructions.

**Default:** None. Unless both `UidFile` and `uid_server` are specified, the UID Server is not used.

**Example:** `uid_server=222.111.111.111:1234`  
`uid_server[Netrust]=222.111.111.111:1111`  
`uid_server[Entrust]=222.111.111.111:1111`

## UserInfoToWebServer

**Purpose:** Specifies whether the SmartGate end user's information is to be sent to the Web server and, if so, the method and format to be used in sending the information. For any information to be sent to the Web server, it must be listed as a `SmartGate_aware` service.

**Format:** `UserInfoToWebServer=value`

**Value(s):** *value* may be one of the following:

**NO** The SmartGate user's information will not be sent to the Web server. **NO** is case insensitive.

**MSG\_BODY** The SmartGate user's information will be:

- appended to the URL
- placed at the beginning of the body of the Web message, provided the message method is POST or PUT.

**YES** The SmartGate user's information will be appended to the URL, and placed in a separate Web message header line starting with **SMARTUSER** if the web server is identified as a SmartGate aware entry in `sgconf.ini`. The user's information can be retrieved by a CGI script from the environment variable **HTTP\_SMARTUSER**. **YES** is case insensitive.

**name** The SmartGate user's information will be:

- appended to the URL
- placed in a separate Web message header line starting with *name*. In this way the environment variable becomes **HTTP\_name**.

**Formats:** The following are examples of the different ways that the user information will be formatted when sent to the Web server.

- If the user's information is appended to the URL:  
`?SMARTUSER=user&SMARTGROUP=group&LONGNAME=longname&IP=IPaddress`

- If the user's information is in a separate Web message header line:

*SMARTUSER: user&group&longname&IPaddress&*  
or

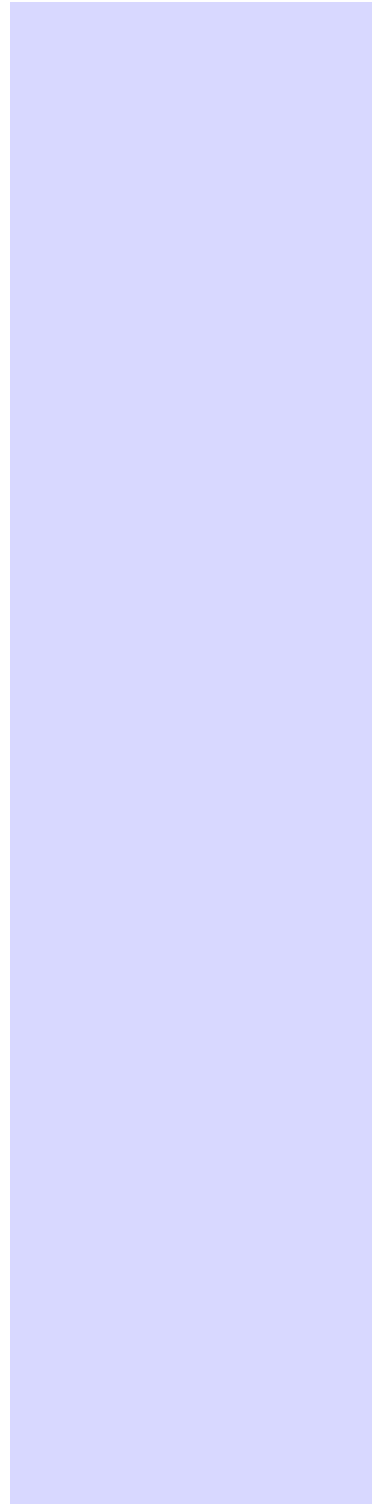
*name: user&group&longname&IPaddress&*

- If the user's information is in the body part of a Web message:

*SMARTUSER=user&SMARTGROUP=group&LONGNAME=longname  
&IP=IPaddress&*

where *user* is that user's User ID  
*group* is the name of the group assigned to that user  
*longname* is that user's long name (the first 2 fields  
from the OLR User Information Window)  
*IPaddress* is the IP address of that user's computer  
*name* is given above

**Default:** None. If this option is omitted from `sgconf.ini`, this feature is not used.



# Appendix B

## Services

This appendix describes optional monitoring services available with SmartGate and applications that facilitate administering SmartGate.

### Accounting Service

SmartGate allows you to log usage data packets from your SmartGate Server. When activated, the usage data packets are sent at the end of each SmartGate session.

To activate the Accounting Service on your SmartGate Server, enter the hostname or IP address and port number of the accounting service in the **Accounting service host and port** text box.

The data is a UDP packet containing eight fields that are delimited by ampersands (&) (see example below). It is formatted as follows:

1. IP address of SmartPass or client proxy—for example, 10.0.0.10.
2. User ID of SmartPass—for example, joesmith.
3. Session start time in seconds of elapsed time since 00:00:00 GMT, January 1, 1970—for example, 831135001.
4. Session stop time in the same format as session start time—for example, 831135032.
5. Destination server hostname (or IP address) and port number, or URL—for example:  
/www.xyzco.com:80/gold/goldlogo.gif
6. Number of bytes transferred from the SmartGate Server to SmartPass—for example, 2578.
7. Number of bytes transferred from SmartPass to the SmartGate Server—for example, 87.

**NOTE:** Using SmartAdmin, select the **Configuration** tab, and then the **Logging** button.

**NOTE:** The configuration setting for the Accounting Service is `accounting_service` located in `sgconf.ini` in the SmartGate Server's root directory/etc on a UNIX-based server and SmartGate Server's root directory\data on Windows NT.

**NOTE:** Maximum URL length is 256 characters.

8. For future use, the currently hard-coded misc.

A packet might therefore show the following information for a Web service:

```
10.0.0.10&joesmith&831135001&831135032&/www.xyzco.com:80/gold/
goldlogo.gif&2578&87&misc
```

No capture or reporting capabilities are provided. However, if you want to take advantage of this feature, a sample capture program, `udp_rd.c`, is displayed below.

### Example:

```
/*
 * This is a sample program that receives accounting record in UDP datagram.
 */
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <stdio.h>

#define PORT_NBR 2060

void main(void)
{
    int n, sd, length;
    struct sockaddr_in name;
    char buf[1024];

    /* create a socket for reading */
    sd = socket(AF_INET, SOCK_DGRAM, 0);
    if (sd < 0) {
        perror("opening datagram socket");
        exit(1);
    }
    name.sin_family = AF_INET;
    name.sin_addr.s_addr = INADDR_ANY;
    name.sin_port = htons(PORT_NBR);
    if (bind(sd, (struct sockaddr *)&name, sizeof(name)) != 0) {
        perror("binding datagram socket");
        exit(1);
    }
    length = sizeof(name);
    if (getsockname(sd, (struct sockaddr *)&name, &length) != 0) {
        perror("getting socket name");
        exit(1);
    }
    printf("Socket has port# %d\n", ntohs(name.sin_port));
    while (1) {
        if ((n=read(sd, buf, 1024)) < 0) {
            perror("receiving datagram packet");
            break;
        }
        buf[n] = '\0';
        printf("-->(%s)\n", buf);
    }
    close(sd);
}
```



## Denial Server

In order to provide greater customer flexibility for implementing access control, a “hook” has been implemented allowing customer-provided servers to override SmartGate Web access control.

**NOTE:** The Denial Server can run on any machine accessible via TCP and you can use any Denial Server that uses that protocol.

**NOTE:** Using SmartAdmin, select the **Configuration** tab, and then the **Access Control** button.

**NOTE:** The configuration setting for the Denial Server is `denial_server` located in `sgconf.ini` in the SmartGate Server’s root directory/ etc on a UNIX-based server and SmartGate Server’s root directory\data on Windows NT.

Written by the customer, the Denial Server process follows a preset protocol reporting back to SmartGate if a Web access request should be granted or denied. Because the Denial Server is checked before the Web Proxy (`sweb`), it is capable of overriding those permissions. The Denial Server listens on a specified port, receives details on Web requests, and “decides” whether the requests should be granted, denied, or passed on for checking against the permissions assigned in `sweb.ac1`.

If you want to write your own Denial Server process to deny access to specific User IDs, you must:

1. Configure the **Web Denial Server host and port** setting so that it points to your process. If a port number is assigned but no host, then the host will default to `127.0.0.1` (`localhost`).
2. Create a TCP/IP service and run it.

A TCP socket connection is made to the service and the Web access data is sent to that process. The process should return a single-word response.

The following protocol is used between `sweb`, the SmartGate Server process, and your Denial Server:

- The process is sent the Web access data as a string containing *name=value* pairs delimited by new lines in the format:

```
UID=user_id\nURL=requested_url\nGROUP=group_name\nNAME=user_long_name\nREGTIME=timestart\nREGUTIME=timeval\nEND\n
```

For example:

```
UID=John12345\nURL=/www.v-one.com/\nGROUP=Sales\nNAME=John Smith\nREGTIME=06/18/1999 09:25:38\nREGUTIME=945938745\nEND\n
```

where: REGTIME and REGUTIME are the date and time of user registration.

*timestart* is the date and time in the format;

“mm/dd/yyyy hh:mm:ss”.

*timeval* is a decimal number representing seconds since January 1, 1970, GMT (UNIX time).

\n is a new line.

- The Denial Server protocol is permitted one of three new line-delimited, single-word responses to `sweb`:

1. GRANT Grant the request
2. DENY Deny the request
3. PASS Pass the request on for `sweb.acl` check (i.e., no judgment)

On receipt of the response, SmartGate will close the connection and proceed with processing. Any failure during this process (e.g., failure to connect to the Denial Server, a time-out during exchange with the Denial Server, or an unrecognized response from the Denial Server) will default to DENY.

## Switching Between the Production and Debug Logs

Only essential production messages are logged in production mode, which is the default mode. Essential production messages include error messages and user information messages. Additional troubleshooting messages are logged in debug mode. The production and debug logging information is written to the Microsoft Windows NT Event Viewer or to `/var/log/smartgate` on a UNIX-based SmartGate Server.

To switch to debug mode from production mode, change **debug=0** to **debug=*n*** (where *n*>0) in `sgconf.ini`, using SmartAdmin. The effect on logging is immediate and does not require restarts or reboots of any kind. To switch back to production mode, simply reverse the procedure—change **debug=*n*** to **debug=0**. As of SmartGate Server 4.0, you can also decrease the amount of information being logged in production mode to only error messages by setting **debug=-1**. However, the -1 setting should only be used if absolutely necessary as it decreases your SmartGate logging information considerably. It increases performance at the expense of security.

**NOTE:** On a UNIX Server you should already have created an administrative user with read/write privileges during installation of the operating system. You will need your username and password.

**NOTE:** You must have user read/write privileges on an operating system to transfer files using FTP.

**WARNING!** You must be in binary mode—NOT ASCII MODE!!

## How To FTP

FTP (File Transfer Protocol) is a way of moving one or more files from one computer to another on the same network. It is especially useful when the files are too large to fit on a floppy disk or when moving files between different operating systems.

To FTP (send) a file from your Windows machine to a UNIX system:

1. Open an MSDOS command prompt.
2. If you have multiple partitions, you may need to change the partition letter first by typing:

***partition:***

where: *partition* = the letter of the partition (i.e., C, D, etc.)

3. Change to the directory where the file you want to send resides.

***cd /directory***

where: *directory* = the directory where the file resides

4. Open an FTP connection to the UNIX Server by typing:

***ftp hostname***

where: *hostname* = the hostname or IP address of your UNIX Server

5. Type your UNIX username and the password.
6. Change the directory on the UNIX Server by typing:

***cd /usr/smartgate/etc***

where: */usr/smartgate* = is the SmartGate Server's root directory

7. Change to binary mode by typing:

***bin***

8. Send the file you want to transfer to the Unix Server by typing:

***put filename***

where: *filename* = the name of the file to be transferred

9. You may also transfer a file residing on the UNIX Server back onto your Windows computer from the same MSDOS command prompt using the **get** command. Type:

**get *filename***

where: *filename* = the name of the file to be transferred

10. Exit FTP mode from the MSDOS command prompt by typing:

**bye**

**NOTE:** When you transfer a file using FTP, the file is transferred from and to the directories on the two machines where you are located at that time. If you type in a file name and the system cannot find it, check that you are in the correct directory on that system. You may need to exit FTP and change directory again using the **cd** command, or use the “**lcd**” (lowercase L) command while using FTP .

# Appendix

## C

**NOTE:** The `vplug` proxy does not support wildcarding.

**NOTE:** Wildcarding does not work without the shim or IPSec drivers being installed.

# ACL Wildcarding

Wildcarding as it applies to SmartGate is the ability to specify a group of Internet destinations or ports with a single rule. In previous versions of SmartGate when a user attempted to access a specific host, a direct match was done; with Access Control List (ACL) wildcarding, SmartPass and the Winsock shim do wildcard matching instead. Wildcards are permitted in the host field, in either hostname or IP address formats, and in the port field. ACL Wildcarding is available on:

- All Windows NT and UNIX operating systems currently supported by SmartGate 2.7 and later.
- All platforms supported by SmartPass 3.4 and later: Macintosh and Microsoft Windows 95/98, NT and CE.

SmartGate's wildcarding abilities can be divided into three distinct types:

- Hostname (DNS) wildcarding
- Address (IP) wildcarding
- Port wildcarding

Wildcarded hostnames, IP addresses and ports can be entered through SmartAdmin (or placed in `sweb.acl` and `sgate.acl`) in place of standard addresses.

## DNS Wildcarding

DNS wildcarding allows the system administrator to create access permissions to a group of computers “behind” the SmartGate Server that share a common DNS domain. For example the wildcarded rule “\*.yourcompanyname.com” would allow incoming client requests to connect to computers with the following DNS names:

```
yoda.yourcompanyname.com
null.yourcompanyname.com
release.yourcompanyname.com
```

The wildcarded rule “\*.yourcompanyname.com” would **NOT** allow incoming client requests to connect to computers with the following DNS names:

```
picket.fence.yourcompanyname.com
si237.inside.fence.yourcompanyname.com
yourcompanyname.com
```

In short, The wildcard character “\*”:

1. Will match one and only one DNS label
2. Can only be stacked from the left of a wildcarded address
3. Cannot make up the entirety of the wildcarded address
4. Must be contiguous

## Valid and Invalid DNS Wildcarded Addresses

The following DNS wildcarded addresses **are valid**:

```
*.yourcompanyname.com
*.*.*.*.*.fence.yourcompanyname.com
*.*.si237.fence.yourcompanyname.com
```

The following DNS wildcarded addresses **are invalid**:

```
*
*.*
*.*.*.*.*.*.*.*.*.* (etc)
*.yourcompanyname.*
si237.*.yourcompanyname.*
*.*.yourcompanyname.*
www.yourcompanyname.*
```

It is *very* important to understand that the correspondence between DNS addresses and IP addresses is **not 1:1**. For instance, the computer bob.johnson.com could have IP address

1.2.3.4, and the computer joe.johnson.com could *also* have IP address 1.2.3.4. In addition, it is possible to create wildcarded DNS rules that *overlap* wildcarded IP address rules, that is, to create a case where both rules specify *different* permissions to the *same* computer. Great care must be taken by the system administrator when using DNS-level permissions, especially when done in conjunction with IP level permissions. Wildcarding gives the administrator the ability to create many types of overlapping permissions. The behavior of SmartGate in the case of overlapping permissions is undefined—it may secure one of the overlapping paths, or none. Wildcarding allows for greater flexibility but great care must be taken not to use this flexibility to create wildcarded rules that overlap each other. If you are not familiar with the IP and DNS naming systems it is strongly recommended that you consult a system administrator before implementing DNS based wildcarding.

## IP Wildcarding

IP wildcarding allows the system administrator to create access permissions to a group of computers “behind” the SmartGate Server that share a common set of bits in their IP addresses. The most common use of IP address wildcarding would be to specify permissions to a subnet. Three types of IP wildcards are supported by SmartGate 2.7 and SmartPass 3.4 and later versions.

### Subnet-Mask IP Wildcarding

Subnet-mask IP wildcarding allows the system administrator to specify which bits are significant in an IP address. A subnet-mask IP wildcard is made up of an IP address in dotted-four form, followed by a colon, followed by a subnet mask in dotted-four form. For example:

```
129.2.179.1:255.255.255.0
```

In this example, the subnet mask “255.255.255.0” specifies that only the first three bytes of the IP address are significant. Since a byte is 8 bits, this is the same as saying that the first 24 bits of the IP address are significant. Therefore, this wildcard would allow all IP addresses starting with “129.2.179” to gain access rights. Another example:

```
129.2.179.50:255.255.255.128
```

The subnet mask “255.255.255.128” in binary is 25 “1” bits followed by seven “0” bits, for a total of 32 bits (the length of an IP address). This specifies that only the first 25 bits are significant. In the IP address portion of the wildcard, the IP address “129.2.179.50” does not have its 25<sup>th</sup> bit set to 1. Therefore this wildcard will allow all IP addresses starting with “129.2.179” whose fourth byte is less than 128 to gain access rights.

Any two valid IP addresses separated by a colon make up a valid subnet-mask IP wildcard.

## Significant-Bit IP Wildcarding

Significant-bit IP wildcards basically specify how many bits from the left of an IP address are significant. For example:

```
129.2.179.100/27
```

Tells SmartGate to match the first 27 bits of “129.2.179.100” against all incoming IP addresses. The equivalent way of specifying the *same* rule in subnet-mask form would be:

```
129.2.179.100:255.255.255.224
```

...because 255.255.255.224 is an IP address with the first 27 bits set to “1”.

## \*-Character IP Wildcarding

\*-character IP wildcarding allows the system administrator to replace one or more bytes of the IP address with “\*”. This will tell SmartGate to allow any value for the byte replaced with “\*”. For example:

```
129.2.*.*
```

is the same as:

```
129.2.0.0\16
```

which is the same as:

```
129.2.0.0:255.255.0.0
```

As you can see, \*-character IP wildcards are merely a convenient notation for specifying IP address wildcards. They are interchangeable with the other two forms.

## Valid and Invalid \*-Character IP Wildcarded Addresses

The following \*-character IP wildcards **are valid**:

```
129.2.179.*
```

```
129.2.*.*
```

```
129.*.*.*
```

The following \*-character IP wildcards are **invalid**:

```
*.*.*.*
```

```
129.2.*.100
```

```
129.*.179.100
```

```
*.179.2.100
```

```
*.*.179.100
```

```
129.*.*.100
```

These are the only 3 configurations available for \*-character IP wilcards (i.e., 1, 2, or 3 “\*”s).



## Port Wildcarding

In addition to specifying destination ports, SmartGate allows the symbol “\*” to specify *all* destination ports.

An example `sgate.acl` rule:

```
jwnt.fence.yourcompanyname.com * 2023 2023
```

An example `sweb.acl` rule:

```
/jwnt.fence.yourcompanyname.com:*/ 2080
```

Please note that using a wildcard (\*) as a destination port in a `sgate.acl` rule specifies that all generic TCP connections for the specified host be proxied. This does not apply to connections made to a reserved port. Named services such as Oracle and FTP do not fall under the generic TCP proxy case, and must be specified separately.

**WARNING!** It is possible (*but strongly discouraged!*) to specify both an `sweb.acl` rule and a `sgate.acl` rule to the same destination with “\*” as the port. For example:

```
sweb.acl:    /www.yourcompanyname.com:*/  
sgate.acl:  www.yourcompanyname.com *
```

*Conflicting access permissions of this type may lead to a denial of service to SmartPass users. The behavior of SmartPass when it is given rules containing such a conflict is undefined and may result in users attempting to create connections using the wrong proxy.*

## Order of Evaluation

SmartPass downloads and reads a user’s ACLs during authentication and Dynamic Configuration, creating a list of available access permissions for that end user.

SmartPass checks each access permission for a match using the following method:

1. Checks for an exact hostname (no wildcards) match
2. Checks for an exact port (no wildcards) match
3. Checks for a wildcarded port match
4. Checks for a wildcarded hostname match

**NOTE:** During Dynamic Configuration, if a new access permission is added during a SmartPass session, that permission is displayed at the end of the list of available access permissions in the SmartPass user interface.

**WARNING!** This ordering of evaluation applies separately to ACLs from one SmartGate Server. No ordering applies to rules which may overlap or conflict if those rules are supplied from several servers. This might be the case if your token contains keys for several servers.

# ACL Specification

## TCP Access Permissions:

- The destination host field will be checked for validity based on new syntax for wildcarded hosts
- An asterisk will be permitted in the destination port field

## Web Access Permissions:

The URL field will be checked for validity based on new syntax for wildcarded hosts.

## Grammar for the `sgate.ac1` File

```
<sgate-acl-file> ::= <sgate-acl-groups>
<sgate-acl-groups> ::= <sgate-acl-group> | <sgate-acl-groups> <sgate-acl-group>
<sgate-acl-group> ::= "[" <group-name> "]" "\n" <sgate-acls>
<group-name> ::= "~" <group-label>
<sgate-acls> ::= <sgate-acl-entry> | <sgate-acls> "\n" <sgate-acl-entry>
<sgate-acl-entry> ::= <sgate-acl-rule> | <group-name>
```

## Grammar for the `sweb.ac1` File

```
<sweb-acl-file> ::= <sweb-acl-groups>
<sweb-acl-groups> ::= <sweb-acl-group> | <sweb-acl-groups> <sweb-acl-group>
<sweb-acl-group> ::= "[" <group-name> "]" "\n" <sweb-acls>
<sweb-acls> ::= <sweb-acl-entry> | <sweb-acls> "\n" <sweb-acl-entry>
<sweb-acl-entry> ::= <sweb-acl-rule> | <group-name>
```

## Grammar for One Access Permission Rule in `sgate.ac1`

```
<sgate-acl-rule> ::= ["@" <smartgatehostname> "/" ] <host-mask> <ws> <std-port-entry> | <wild-port-entry>
<std-port-entry> ::= <destport> [<ws> <sp-port> [<ws> <sg-port> ] ]
<wild-port-entry> ::= <wildport> <ws> <sp-port> <ws> <sg-port>
<wildport> ::= "*"
<destport> ::= <number>
<sg-port> ::= <number>
<sp-port> ::= <number>
<ws> ::= <white space char> [ <ws> ]
```

## Grammar for One Access Permission Rule in `sweb.ac1`

```
<sweb-acl-rule> ::= "/" <host-mask> [ ":" <destport> | <wildport> ] [ "/" <url-path> ] "/"
<ws> <sg-port>
```

## Grammar Details for Handling Wildcards in ACLs

What the following grammar is saying is that a wildcarded destination may be either a wildcarded domain name or a wildcarded IP network. A wildcarded domain name can only contain leading asterisks. A wildcarded IP network can be either an IP address, and IP address with trailing asterisks, an IP address followed by a colon and a subnet mask, or an IP address followed by a slash and a number of significant bits.

```
<host-mask> ::= <name-mask> | <ip-mask>

<name-mask> ::= <asterisks> "." <name> | <name>
<asterisks> ::= "*" | "*" "." <asterisks>
<name> ::= <label> | <label> "." <name>
<ip-mask> ::= <wild-ip-address> | <ip-address> ":" <subnet-mask> | <ip-address> "/"
<sigbits>

<subnet-mask> ::= <dotted-four>
<ip-address> ::= <dotted-four>
    <dotted-four> ::= <number> "." <number> "." <number> "." <number>
<sigbits> ::= <number>
<wild-ip-address> ::= <ip-address> | <wild1> | <wild2> | <wild3>
    <wild1> ::= <number> "." <number> "." <number> "."
    <wild2> ::= <number> "." <number> "." *
    <wild3> ::= <number> "." *
<group-label> ::= a <label> of up to 23 characters
<label> ::= <let-dig-hyp> | <let-dig-hyp> <ldh-str>
    <let-dig-hyp> ::= <let-dig> | "-"
    <let-dig> ::= <letter> | <digit>
    <letter> ::= any one of the 52 alphabetic characters A through Z in
        upper case and a through z in lower case
    <digit> ::= any one of the ten digits 0 through 9
<number> ::= <digit> | <digit> <number>
<white space char> ::= <space> | <tab>
```

**NOTE:** While upper and lower case letters are allowed in domain names, no significance is attached to the case. That is, two names with the same spelling but different case are to be treated as if identical. Labels as defined here are any sequence of letters, digits, and hyphens. Labels must be 63 characters or less.

## Pattern Matching Actual Destinations with Wildcarded Destination ACLs

When pattern matching a given destination host against a set of <host-mask>s, each label is compared separately. Only when all labels match, is a match considered to be made.

Examples:

```
IsMatch("10.0.0.1", "10.0.0.*") is TRUE
```

```
IsMatch("www.yourcompanyname.com", "*.yourcompanyname.com") is TRUE
```

```
IsMatch("fence.www.yourcompanyname.com", "*.yourcompanyname.com") is FALSE
```

```
IsMatch("80", "**") is TRUE
```

```
IsMatch("**", 80) is FALSE
```

```
IsMatch("x.y.yourcompanyname.com", "**.*.yourcompanyname.com") is TRUE
```

```
IsMatch("x.y.yourcompanyname.com", "**.yourcompanyname.com") is FALSE
```

```
IsMatch("y.yourcompanyname.com", "yourcompanyname.com") is FALSE
```

**NOTE:** "\*" can not match anything that contains a ".".

## Validating ACLs Which May Contain Wildcards

1. Must conform to above specified grammar
2. Numbers in <dotted-four>, <wild1>, <wild2>, <wild3> must be in the range of 0 to 255
3. Numbers in <sigbits> must be in the range of 0 to 32
4. <labels> can not exceed 63 characters

## Dealing with Wildcarded Ports

1. An ACL entry with a wildcard for a port number must have an explicitly stated <sp-port>
2. SmartPass will listen on <sp-port>
3. All connections to the host in that ACL will be redirected to localhost <sp-port> by the wsock32 shim

## A

- Access Code 19, 20, 22, 115
  - days valid 44, 122, 260
  - features 22
- access control
  - Authentication Server 125–130
  - configuration settings 102–104
  - fine-grain 115
  - turn on/off use 103, 278, 282, 289
  - vplug Proxy 171
- Access Control List (ACL) 227–230
- Access failure retry delay 115
- access permissions 126
  - denied 127
  - management 49–50, 82–92
  - sample structure 83–84
  - SmartGate groups 126
  - TCP (sgate.acl) 49–50, 84–89, 126
    - add/edit 87
    - delete 88
  - Web (sweb.acl) 49–50, 89–94, 126
    - add/edit 91
    - delete 91
- AccessCodeDaysValid 122, 260
- Accounting service 109, 287–292
- accounting\_service 109, 260
- ACE/Server 131
- Adapter Security Levels 192–193
- adm-gw.acl 51–52, 75, 242–244
- administrative privileges
  - assigning 51–52, 75–76, 96–97, 242–244
  - levels 52, 73, 97, 243
  - SmartGate administrator
    - duties 82
    - setting yourself up 75–76
  - User ID 31, 97

- aliases. 173, 244
- anon\_reg\_allowed 107, 261
- anonymous registration 107, 261
- AuthEncryptMethod 114, 262
- Authentication client hosts 114
- Authentication encryption methods 114
- authentication key 21, 31, 115, 125
- Authentication Server (sgasrv) 113–115, 125–130, 227–230, 276–277
  - backup server 129–131
  - host setting 113
  - remote setup 113–114, 128–129
  - server redirection 128–129
- authentication token 20
  - Entrust 138–146
  - Netrust 33, 57–58
  - PKI 58, 147–148
  - RADIUS 55–60, 133–138
  - SecurID 54–60, 131–133
- authenticator 111, 261
- Authenticator name 38, 64, 111, 150, 261
- AuthMethod 113, 262

## B

- backup configuration files 59–60
- backup server 114, 129–131, 262–263
  - encryption 115
- backup\_userdb 114, 128, 262, 289
- backup\_userdb\_encrypt 115
- branding options 48, 94–95, 259, 267–269
  - company info 48–49, 95
  - desktop icon 48–49, 95
  - firewall 48–49, 95
- BSD/OS 23

## C

- CD-ROM 61
- certification files 43
- chantype.ini 245
- Configuration Client hosts 118
- Customer ID 32

## D

- daemon 154
- database files 241–242
- dbrw 241–242
- debug reporting 110, 263
  - production vs. debug log 290
- Denial Server 104, 263, 289–290
- denial\_server 104
- digital certificates 147
- Distinguished Name 146–147
- dns\_reverse 109, 264
- domain name 38
- domainname 65, 111, 150, 264
- Dynamic Configuration 19, 20, 103
  - agent 115
  - Server 278–279
    - host and port setting 117
  - remote 117

## E

- e-mail 244
- enable
  - User ID 75
- encryption keyname 43, 108, 251
- encryption methods 112, 114, 262, 272, 279
- end user
  - add to user database 78–79
  - authentication 125–130
  - delete from user database 80
  - edit user database 79–80
  - enabled after OLR 107
  - management 77–82
- entCreate 141
- Entrust authentication 138–146
  - .epf file 144
  - configure SmartGate 143
    - activate Entrust at later time 141
  - Distinguished Name 146–147
  - Entrust software 139–140

- entrust.ini
  - configuration 139–140
- Entrust/Netrust
  - Authorization code 139, 140–146, 144
  - Reference number 33, 139, 140–143, 144
- epf file 141–146
  - generate 141–146
- ethernets 19
- Event log service 110
  - turn off 280
- event\_log 110, 265

## F

- files
  - detailed descriptions
    - Access Control Lists 227–230
    - adm-gw.acl 242–244
    - aliases. 244
    - chantype.ini 245
    - configuration files 242–255
    - database files 241–242
    - dbrw 241–242
    - ipsec.acl 227–228
    - net\_list 254–255
    - netaccess.cf 254
    - reginfo.dat 250–253
    - Rules File 255–257
    - sgate.acl 237–238
    - sgate.dny 238–286
    - sgconf.ini 248–249
    - sgusrdb 241–242
    - sites.acl 231–240
    - sweb.acl 239–286
    - sweb.dny 240
  - reginfo.dat options
    - keyname 108
  - sgconf.ini branding options 94–95
    - OLRAllOutsideFirewall 267
    - OLRCity 267
    - OLRCompanyName 268
    - OLRCountry 268
    - OLREmail 268
    - OLRPhone 268
    - OLRStartArgs 268
    - OLRStartDesc 269
    - OLRState 269

- OLRStreetAddress 269
- OLRWebPage 269
- OLRZipCode 269
- sgconf.ini options 258–286
  - AccessCodeDaysValid 122, 260
  - accounting\_service 109, 260
  - anon\_reg\_allowed 107, 261
  - AuthEncryptMethod 114, 262
  - authenticator 111, 261
  - AuthMethod 113, 262
  - backup\_userdb 114, 262
  - backup\_userdb\_encrypt 115
  - compare with SmartAdmin settings 258
  - debug 110, 263
  - denial\_server 104, 263
  - dns\_reverse 109, 264
  - domainname 111, 264
  - event\_log 110, 265
  - example 249
  - InsidIP 111, 265
  - ipsec\_server\_extrn 191, 265
  - Krakit\_Delta\_Days 123, 266–286
  - max\_quiet\_time 103, 266
  - NAT 191, 266
  - NATNet 192, 267
  - OLRMethod 105, 270
  - online\_reg\_enable 107, 270
  - online\_reg\_service 108, 271
  - PortList 111, 272
  - ProxyEncryptMethod 112, 272
  - radius\_authsrv[1...5] 121, 135, 272
  - radius\_authsrv[1...5]\_secret 121, 135, 273
  - radius\_authsrv[1...5]\_usechap 121, 136, 274
  - radius\_authsrv[1...5]\_waitfor 122, 136, 274
  - radius\_challenge\_timeout 122, 136, 275
  - radius\_ttl 122, 136, 275
  - RETRY\_DELAY 115, 275
  - SDI\_TIMEOUT 123, 133, 276
  - SDI\_TTL 123, 133, 276
  - sgasrv 113, 276
  - sgasrv\_clients 114, 277
  - sgateacl 103, 278
  - sgccsrv 117, 278
  - sgccsrv\_clients 118, 279
  - SGEncryptMethod 112, 279
  - sgevent\_logging 280
  - sgftp\_port\_max 123, 280
  - sgftp\_port\_min 123, 281
  - shim\_permitexe 123, 281
  - SmartGate\_aware 118, 281
  - SmartWebPort 123, 282
  - stat\_server 109, 282
  - sweb\_not\_allowed 104
  - swebacl 103, 282
  - ticket\_to\_web\_server 120, 283
  - TrustedCAList 115, 283
  - UDPPortList 111, 284
  - uid\_server 105, 152–164, 284
  - UidFile 107, 152–154, 284
  - UserInfoToWebServer 119–120, 285
- filtering 89, 92
- find/find next 81
- FIPS Token (FIPS 140-1) 26–27
- firewall navigation 163–164
- FTP
  - How to FTP 291
  - Passive mode (PASV) 165

## G

- group, SmartGate 125
  - access permissions 83–84, 126
  - "all" 86, 127, 198, 202
  - identifier 32, 86, 198, 202
- grouplist 47, 94

## H

- HTML OLR page 156–161
  - sample form 160–161

## I

- inetd 154
- InsidIP 111, 150, 265
- install SmartGate Server software
  - UNIX platform
    - additional command line options 132, 134, 142
- installation
  - SmartGate Server software
    - UNIX system 35–60, 142–143
    - Windows NT system 61–72, 140–143
  - values 31–33

- Intel System 23
- introduction 19–27
- IP address(es) 271
  - inside 38, 65, 111
  - SmartGate Server 31
- IPSec 179–215
  - access permissions 197–203
    - DNS Proxy 202–203
    - Path or Include 198–200
  - Authentication Header 180–183
  - channels
    - add/edit 195–196
    - delete 196–197
  - configuring channels 193–196
  - configuring the SmartGate Server 191
    - Adapter Security Levels 192
    - IPSEC Server External Interface 191
    - NAT Enabled 191
    - NAT Network 192
  - core components 179–183
  - DNS Proxy 200–203
  - Encapsulating Security Payload 180–183
  - functionality 184–185
  - IP payload compression 181–183
  - Network Address Translation 181–183
  - network configurations 185–189
    - adapters (interfaces) 189
  - overview 179–183
  - parallel topology 185
    - adapter settings 193
  - protocol number definitions 205–207
  - serial topology 185
    - adapter settings 192
  - server files
    - chantype.ini 245
    - ipsec.acl 227–231
    - sites.acl 231–240
  - setting TCP/IP parameters 30, 189–190
  - site-to-site 208–215
    - add permissions 211–212
    - configuration information 209–215
    - deploy new keys 214
    - edit permission 213–214
    - export new tunnel 212–213
    - firewall description 208
    - generate new keys 214

- import tunnel 213
  - operational description 208
  - routing description 209
- SmartAdmin tab 209–215
- toaster topology 185
  - adapter settings 193
- transport & tunnel modes 180
- VIPUTIL utility 204
- ipsec.acl 227
- ipsec\_server\_extrn 191, 265

## K

- keyname 108
- keyname.has 251
- keyname.prv 43
- keyname.pub 251
- Krakit\_Delta\_Days 123, 266–286

## L

- licensing 27, 68
  - Certificate 69–70
  - Customer ID 32
  - License Key 69–70
    - viewing 44
  - private key 69–70
  - Serial Number 32
- Linux Systems 23
- logging 108–110
  - debug setting 263
  - Discrete Events Server 265
  - production vs. debug log 290
  - vplug Proxy 171

## M

- Macintosh operating systems (Mac OS)
  - SmartPass requirements 25
- mail program
  - aliases. 244
- mainframes 19
- Manual Setup 103
- max\_quiet\_time 103, 266



## N

- name=value fields 150–151
- NAT 191, 266
- NATNet 192, 267
- net\_list 175, 254–255
- netaccess.cf 172–173, 254
- Netrust authentication 57–58
  - anonymous registration 107, 261
  - Authorization code 33
  - entrust.ini 33
  - OLR Method 105, 270
- new version features 25–26

## O

- OLR Server 111, 151, 255, 264, 271
- OLRAIIOutsideFirewall 267
- OLRCity 267
- OLRCompanyName 268
- OLRCountry 268
- OLREmail 268
- OLRMethod 105, 270
- OLRPhone 268
- OLRStartArgs 268
- OLRStartDesc 269
- OLRState 269
- OLRStreetAddress 269
- OLRWebPage 269
- OLRZipCode 269
- On-Line Registration (OLR) 149–164
  - Activity Recording Service 271
  - Authentication Server 125–130
  - configuring 93–95, 105–108
    - branding information 48, 94
    - user information 45–48
  - data destination 108
  - enabled automatically 107, 270
  - files
    - reginfo.dat 250–253
    - sgconf.ini 45, 248–249
  - firewall navigation 163–164
  - manual setup of HTML page 156–161
    - sample form 160–161
  - UID Server 150–156, 284–285
    - Rules File 255–257, 284–286
- online\_reg\_enable 107, 270

- online\_reg\_service 108, 271
- Oracle SQLNet II Proxy 167–170
- Oracle SQLNet Proxy
  - access permissions setup 169
  - single port configuration 170
  - SmartGate Server setup 167–168
    - Oracle setup 168
  - testing connection 169–170
  - troubleshooting 170
- overview 13–16

## P

- PASSIVE command 165
- Passive Open (PASV) 165–166
- personal certificates 148
- PKI authentication 58, 147–148
  - add CA certificates 148
  - digital certificates 147
  - personal certificates 148
- Pocket PC Devices 25
- PortList 111, 272
- Protocol number definitions 205–207
- proxy
  - vplug 171–177
- Proxy encryption methods 112
- ProxyEncryptMethod 112, 272
- public/private key pair
  - regenerate 42–43

## R

- RADIUS authentication 55–60, 133–138
  - configuring the RADIUS Backend Server 137
  - configuring the RADIUS port 136–137
  - configuring the SmartGate Server 121–122, 135–137
  - installation values 32
  - running sgradius 133–135
  - SmartPass/RADIUS interaction 137–138
- radius\_authsrv[1...5] 121, 135, 272
- radius\_authsrv[1...5]\_secret 121, 135, 273
- radius\_authsrv[1...5]\_usechap 121, 136, 274
- radius\_authsrv[1...5]\_waitfor 122, 136, 274
- radius\_challenge\_timeout 122, 136, 275
- radius\_ttl 122, 136, 275
- reginfo.dat 161, 250–253, 253
  - configuring 45–48, 93–94

- remote administration
  - minimal UNIX configuration 41–42
  - SmartAdmin preliminaries 75–76
- Remote Authentication Server 128–129
- RETRY\_DELAY 115, 275
- Reverse DNS lookups 109
- RFC959 standard 165
- root administrator 31
- Rules File 107, 152–156, 255–257, 284–286
  - guidelines 153–154, 256
- run time libraries 57

## S

- script command 36
- sdconf.rec 131
- SDI\_TIMEOUT 123, 133, 276
- SDI\_TTL 123, 133, 276
- SecurID authentication 54–60, 131–133
  - configuring the SmartGate Server 123, 133, 276
  - making SmartGate a client 131
  - running sgstdi 131–132
- Server proxy timeout 103
- Server redirection rules 128–129
- Servers
  - ACE 131
  - Authentication 113–115, 125–130, 227–230, 276–277
  - Denial 104, 263, 289–290
  - Dynamic Configuration 278–279
  - OLR 151, 264
  - RADIUS Backend 137
  - User ID (UID) 105–107, 150–156, 284–285
  - Web 104, 157–160, 239–240, 285–286
- setup.ini
  - installation packaging 144–146
- sgasrv 113, 125–130, 276
- sgasrv\_clients 114, 277
- sgate 278
- sgate.acl 49, 49–50, 126, 237–238, 278–286
  - server redirection rules 129
- sgate.dny 127, 238–286
- sgateacl 103, 278
- sgccag 115
- sgccsrv 115, 117, 278
- sgccsrv\_clients 118, 279
- sgconf.ini
  - configure using
    - SmartAdmin 102–123
    - UNIX install script 44–45, 48
  - detailed description 248–249
  - example 249
  - option descriptions 258–286
  - RADIUS settings 135–137
  - SecurID settings 133
- SGEncryptMethod 112, 279
- sgevent\_logging 280
- sgftp 278
- sgftp\_port\_max 123, 280
- sgftp\_port\_min 123, 281
- sgkeys 120, 283
- sgora 278
- sgproxy.conf 53, 98, 170
- sgradius service 133–135
- sgreg.usr 108
- sgstdi service 131–132
- sgstdsrv 105, 152
- sgusrdb 241–242
- shim\_permitexe 123, 281
- Single Port client 97
- Single Port Proxy
  - change default proxy 100–101
  - configuring 53, 98–101
  - default port map 99
- sites.acl 231–240
- smart cards
  - authentication 22
  - physical 19, 21, 126
  - virtual 19, 21, 126
  - FIPS token 26–27
- SmartAdmin 73–124
  - administrative privileges 96–97
  - basic usage 76
  - configuration options 102–123
    - access control settings 102–104
    - authentication settings 113–115
    - compare with sgconf.ini file names 258
    - destination configuration settings 118–120
    - dynamic configuration settings 115–118
    - IPSec settings 191, 209–215
    - logging settings 108–110

- OLR settings 105–108
  - other settings 122–124
  - RADIUS settings 121–122
  - system definition settings 111–112
- managing users 77–82
- remote administration 75–76
- setup 74–76
- Single Port Proxy 98–101
- TCP access permissions 84–89
- Users Table 77
- Web access permissions 89–94
- SmartGate encryption methods 112
- SmartGate Server 21, 276
  - files 227–286
    - detailed descriptions 227–257
    - option descriptions 258–286
  - hardware/software requirements 22
  - IP address(es) 31, 111
    - inside 265–286
  - new features 25–26
  - Oracle SQLNet Proxy 167–170
  - preinstallation 29–33
    - setting TCP/IP protocol properties 30, 189–190
  - Single Port Proxy 98–101
    - change default 100–101
  - user database 21, 77
  - vplug Proxy 171–177
- SmartGate Server software
  - installation
    - values 31–33
  - UNIX installation 35–60
    - administrative privileges 51–52
    - backing up configuration files 59–60
    - configure sgconf.ini 44–45
    - Entrust authentication 142–143
    - extensible components 54–60
    - general instructions 36–40
    - installation script 36–39
    - licensing 44
    - Main Menu 37
    - Netrust authentication 57–58
    - OLR branding 48
    - OLR user information 45–48
    - PKI authentication 58
    - RADIUS authentication 55–60
    - remote administration 41

- SecurID authentication 54–60
    - set up system 42–58
    - single port proxy 53
    - start installing 38
    - upgrade 35
  - Windows NT installation 61–70, 61–72
    - adding/removing services 71, 168
    - administrative privileges 75, 76
    - licensing 69–70
    - upgrade 61–70
- SmartGate services 287–292
- SmartGate System
  - Access Code 22
  - components 20–22
  - IPSec 179–183
    - network configurations 185–189
  - smart card authentication 22
  - SmartPass 21
- SmartPass
  - new features 26
- SmartPass (Macintosh)
  - "Macintosh USERS" notes 21, 167
  - OLR Web page
    - branding options 48
- SmartPass 4.x
  - change Single Port Proxy default 100–101
  - end user management 45–48, 77–82
  - prepare installation package
    - Entrust authentication 144–146
  - prepare On-Line Registration 93–95
    - manual setup of HTML page 156–161
    - OLR Web page branding 48, 94–95
  - Single Port client 97
- SmartPass software 21
  - hardware/software requirements 24–25
  - Pocket PC devices 25
  - Windows CE devices 25
- SmartWebPort 123, 282
- SSL Proxy 163
  - requiring authentication 163
- stat\_server 109, 282
- Sun SPARC System 23
  - SmartPass for Linux 24
  - SmartPass for Solaris 24
- sweb 104
- sweb.acl 49, 49–50, 126, 239–286
  - server redirection rules 129

sweb.dny 127, 240  
sweb\_not\_allowed 104  
swebacl 103, 282

## T

TCP/IP 24  
    interoperability 19  
    setting protocol properties 30, 189–190  
ticket\_to\_web\_server 120, 283  
token rings 19  
Triple DES encryption  
    AuthEncryptMethod 114, 262  
    ProxyEncryptMethod 112, 272  
    SGEncryptMethod 112, 279  
Trust CA list 115  
TrustedCAList 115, 283

## U

UDPPortList 111, 284  
UID Server (sguidsrv) 41, 51, 93, 150–156  
    configure SmartGate Server 151–154  
    create your own process 155–156, 289–290  
    deny OLR access 156  
    register 154  
    Rules File 107, 153–164, 255–257  
    setup 105–106  
    setup checklist 151–154  
uid\_server 105, 152–164, 284  
UidFile 107, 152, 255, 284  
uninstall  
    UNIX SmartGate Server software 60  
UNIX environment  
    SmartPass requirements 24  
UNIX SmartGate Server  
    additional command line options  
        132, 134, 142  
    Entrust authentication 142–143  
    registering UID Server 154–164  
    UID Server setup 57  
Usage service 109  
user authentication 125–148  
    authentication key 31  
    Entrust 138–146  
    PKI 147–148  
    RADIUS 133–138  
    SecurID 131–133

SmartGate Authentication Server 125–130  
user database (sguserdb) 21, 77, 241–242  
    dbrw application 241–242  
    redundant 129–131  
User ID 31, 77, 78, 86, 93, 125, 198, 202  
    generating with UID Server 105–107, 150–156  
User info to Web services 119–120  
UserInfoToWebServer 119–120, 285  
user's long name 47, 77, 93, 125, 252, 252–253

## V

VIPUTIL utility 204  
vplug Proxy 171–177  
    net\_list guidelines 175, 254–255  
    netaccess.cf 172–173, 254  
Windows NT System 176

## W

Web OLR page  
    branding options 48, 94–95  
    manual setup of HTML page 156–161  
Web Proxy (sweb) 104, 163, 289  
    requiring authentication 163  
Web server 104, 157–160, 239–240, 285–286  
Web services requiring encrypted tickets  
    120, 283  
wildcards 82, 84, 92, 175  
Windows CE devices 25  
Windows environment 19  
    SmartPass requirements 24  
Windows NT System 24  
    adding/removing services 71  
    Entrust authentication 140–143  
    establishing SmartGate administrator 75, 76  
    hard disk partition 61  
    installing server software 61–70  
    InstallShield 62  
    running sgradius service 168  
    SmartPass requirements 24  
    UID Server 154  
    upgrading server software 61–70  
Winsock function call interception 281  
wsock32 281